

NETWORK PUBLISHING WITH WIDE AREA INFORMATION SERVERS

Brewster Kahle

Richard Koman

7/30/94

DRAFT of first 4 chapters

PLEASE DO NOT DISTRIBUTE

TABLE OF CONTENTS

Network Publishing with.....	1
Table of Contents.....	1
Network Publishing.....	1
Emergence of Network Publishing.....	2
Elements of Network Publishing.....	4
Technology for a New Industry.....	11
Clients: Explorers, Searchers, Agents.....	14
WAIS in Business: A Scenario.....	15
Searcher's Interfaces (Sidebar).....	20
Agents and Personal Newspapers.....	20
Gateways.....	26
Conclusion.....	33
Chapter 3: The WAIS Server.....	35
Introduction.....	35
Becoming a Network Publisher.....	36
Inside the WAIS system.....	38
Other Server Issues.....	56
How does WAIS compare to other search engines?.....	58
The WAIS Protocol Suite:.....	64
Requirements of a Network Publishing Protocol.....	65
Protocol Session.....	68
Interoperability on the Internet.....	70
Overview of the Protocol Suite.....	75
Evolving Aspects of the Protocol.....	79
Future Directions of the Protocol.....	81
Chapter 5: WAIS Systems.....	81
Chapter 6: Future Directions.....	81
Acknowledgements.....	81
Index.....	81

CHAPTER 1

NETWORK PUBLISHING

"He who first shortened the labor of copyists by the device of movable types was disbanding hired armies, and cashiering most kinds and senates, and creating a whole new democratic world."

Thomas Carlyle, 1836

Few generations experience the birth of a technology that explodes the boundaries of communication. The printing press allowed for widespread dissemination of authored works, while the telephone fostered interaction between individuals. Of like importance is the development of structured interaction over computer networks. This new structure allows for many more people to distribute work, while providing readers with powerful navigation tools to move through all this new information. This system, which combines the worlds of the printed book and the switched telephone system, has been called "network publishing."

In a technology shift of this nature, industries are formed, methods of learning and communicating are altered, societies are transformed. Fundamentally, the human experience is forever changed.

With the development of the printing press in the late 15th century, languages became standardized, people tapped the knowledge of the ancients and, more importantly, authors were able to spread their words far and wide. New types of writing flowered and literature was born.

More recently, the telephone connected people in distant locations and allowed for the physical separation of offices and factories. While only businesses had access to the technology in the late 19th century, by the 1930s even rural homes were connected.

Network publishing, as well, has great potential to change -- and improve -- the flow of information. Network publishing opens doors for more authors to inexpensively distribute their works, for those works to be presented in a rich environment that features many kinds of media, for readers to quickly and accurately navigate through this wealth of information -- all within a mix that includes publishing, education, entertainment and commerce.

The early versions of network publishing systems are in place today on the Internet. Although more development is required, technologies like the Wide Area Information Servers, World Wide Web and Gopher systems are proving their worth in this environment. In this chapter, we will look at the trends that make network publishing viable and discuss how today's systems are being built.

Emergence of Network Publishing

Network publishing allows for inexpensive reproduction, targeted transmission and distributed control. As the phrase suggests, network publishing comes out of the idea of a convergence of publishing and networks. The publishing industry is following a trend towards computerization, in which all elements of production are digital until the work is printed. Computer networks, notably the Internet, are now fast, inexpensive and highly distributed. This convergence of content and distribution lays the stage for network publishing. In the following sections, we will explore this convergence in more depth.

Figure 1. The widespread digitization of information, the growth of the Internet, a drop in the cost of distributing information over the network, and increasing costs associated with printing all serve to promote the emergence of network publishing.

Publishing: Computerized Production

The publishing industry has undergone a series of technology shifts, most of which have come from the application of computer technology to such tasks as typesetting. The desktop publishing revolution -- based on Apple Macintosh computers, the Adobe PostScript page description language and graphics software -- accelerated the process.

To understand the impact of this (and how it relates to network publishing), consider how the print publishing process has changed. In any publishing operation, there are several pieces that must be integrated together to create a page that can be printed. The content must be written, the pages must be designed, the words must be typeset, black and white photographs must be converted to halftones, color photographs must be separated into sets of film. Then all the pieces are put together, film is shot of each page, the films are stripped together in a signature, a plate is made from the film, and finally the piece is printed.

The Macintosh and PostScript changed all that forever; entire professions were replaced in the process. With authors writing on computers and designers creating pages on screen, typesetting became inseparable from design. It was simply sucked into the machine. As the technology improved, more and more of these processes -- traditionally performed by skilled tradesmen -- were pulled into the computer as well.

With current technology, all "prepress" functions can be done on desktop computers. But eventually the work is output to film for stripping, plate making and printing on paper.¹

¹ Hand in hand with the computerization of prepress has come a restructuring of the economics of publishing -- up to the point where film is output. From here on out, the economics of publishing remain the same, with publishers incurring substantial costs associated with paper, printing, warehousing, shipping, mailing and waste.

Figure 2. Desktop publishing introduced digital technology to the print production but the end result was still the printing of ink on paper.

Now that documents are created and stored on computers, a natural step is distribute them in digital form. This requires a reliable digital transport that is inexpensive enough, fast enough and widespread enough to support text and graphics. The Internet has recently achieved these goals and is starting to be used as a distribution mechanism for network publishers.

Computer Networks: Fast, Cheap, Distributed

The cost of digital communications has dropped, reflecting the drop in cost in terminal equipment and the rise in demand. While phone lines are used to transmit fax pages, the slow speed means that users tend to receive these documents in the background for later reading. Internet speeds, on the other hand, are seven to 40 times faster, which is fast enough for users to browse, search, and scan text and business graphics. As the speeds go up, audio, video and interactive services will be practical, but for now, network publishing is centered on text and graphics.

With a inexpensive system that is fast enough, the remaining question is: Does the network connect enough users? In 1994, the Internet Society estimated that some 20 million people use Internet e-mail and over a million users are interactively connected, with usage doubling every nine months.²

While the academic and research communities make up the current base of Internet users, the greatest growth is in the international and commercial sectors.³ For publishers that want to address the markets connected to the Internet, that is enough coverage to support the first businesses.

Therefore the Internet is a viable distribution model for network publishing and it (or other similar networks) will likely prosper. The history of the Internet -- from Defense Department research project to education-and-research network to commercial backbone -- is the story of the successful migration of a scalable technology.

Applications on the Internet include games, Electronic Document Interchange (EDI), X.25 replacement and publishing.

Network Publishing

Network publishing will turn many kinds of organizations into international publishers. Everyone who is in the business of providing information to an audience, whether for free or for some payment, may become a network publisher: government agencies, corporations, libraries, individual writers and artists, as well as publishers of magazines, newspapers and books.

² "Internet Usage Statistics," April 1994. The Internet Society, Reston, Va.

³ *ibid.*

The first wave of network publishers -- and this is already happening -- is comprised of organizations who are looking for less-expensive ways of distributing free information. These include corporations, government agencies, university libraries and catalog publishers. Sun Microsystems is an example of this kind of network publisher. Sun uses the Internet to distribute technical marketing materials at a much lower cost and with greater timeliness than could be done by printing and mailing these materials.

A second wave is comprised of newspaper, magazine, journal and book publishers who are "republishing" their content for networks. Dow Jones and Encyclopædia Britannica are two "traditional" publishers intent on making money by network publishing. Because traditional publishers already have in place a system for gathering, editing and presenting information, publishers can easily "repurpose" data collected for the primary business function.

While many publishers are using online services like CompuServe and America Online to publish electronically, these companies are publishing directly on the Internet in order to maintain the profitability of their network publishing business. "The main reason we are doing it ourselves is that you just can't make any money licensing your content," Joseph J. Esposito, president of Encyclopedia Britannica North America said in a *New York Times* article on the Britannica service.⁴ "If you do believe that content is king, it's rather unfortunate that so many of the content providers have put themselves in a position where they're held hostage to the online services."

The third wave will occur when works are created directly for this interactive environment. People will take advantage of the fact that anyone can be a publisher: all that is needed is a computer, a telephone and something to say. Individuals will share ideas and work with others in a way not possible before. Network publishers will be able to find an audience and readers will be able to find compelling documents.

Elements of Network Publishing

A network publishing system is based on client-server technology. In this architecture, a "smart" remote server manages the collection of documents, while users run graphically rich client programs on their personal computers. The two communicate with each other using standard protocols.

The goals of a network publishing system are to provide the user with many different ways to access information, with users determining which is most appropriate for their needs; to have a very large, heterogenous collection of information; and to have the system available to millions of users, by relying on standardized network protocols, not proprietary ones.

⁴John Markoff, "Britannica's 44 Million Words Are Going On Line," *The New York Times*, February 8, 1994

Client-Server Technology: The Power of the Desktop

Client-server frees users from the shackles of the mainframe. They can interact with servers using desktop computers, laptops, palmtop devices, and, maybe someday, home game machines. Client-server technology puts the control in the user's hands, by exploiting the power of the user's computer to provide more functionality and efficiency.

The chief advantages of client-server for network publishing are graphical user interfaces (GUIs), integration with other applications and advanced display modes.

Graphical user interfaces. GUIs typically feature icons and windows, thus hiding complexity and increasing ease-of-use and efficiency. In a client-server environment, the local computer controls the user experience and the server provides fixed services. By contrast, in a remote windowing system, the server controls the entire user experience. Examples of this are America Online and Mosaic.

Ease of use. Applications using icons and menus have been shown to be 35% faster to use than a similar character-based program. GUI users were less fatigued and were found to explore and teach themselves the capabilities of the application.⁵

Integration with the Desktop. Client-server allows users to bring external information into other local programs such as word processors, spreadsheets and image editors. It is also possible to add searching functions directly into these programs, so, for instance, a writer could search for a specific document, download it, display it and edit it, all without leaving the word-processing program. Or a designer could search the network for an appropriate stock photo and add it to the design, all within one page layout program.

Clients: Navigating Through Information

Readers need to be able to search and find the information they need without being exposed to information they don't need, to browse and explore other information when they have the time and inclination, and stay up-to-date by having new information delivered.

A network publishing system should allow users to navigate in a variety of ways, freely choosing between methods as needs dictate. This is not unlike the way we use printed books, with their tables of contents, indices and pages.

⁵ Dekkers L. Davidson. 1990. *The Benefits of the Graphical User Interface*. Temple, Barker & Sloane.

Table of Contents: Browsing

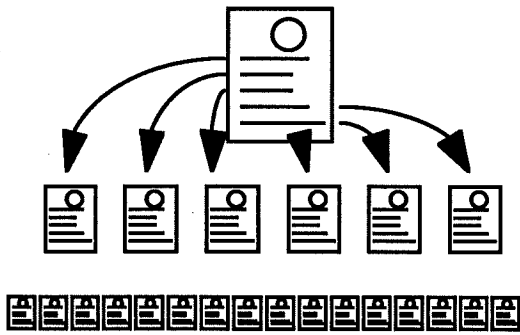


Figure X. Hierarchical category systems guide users to information.

Just as a book's table of contents gives the first indication of what a book is about and how it is organized, hierarchical browsing structures give the online user a sense of the kind of information available in the collection. By categorizing and subcategorizing, the network publisher can provide detailed information about the content to be found within.

On the Internet, Gopher, FTP and the World Wide Web let users browse through menus or directories to find information. Because the World Wide Web supports graphics, authors can use it to create graphical entry documents into a collection of information.

Pages: Hypertext and Graphics

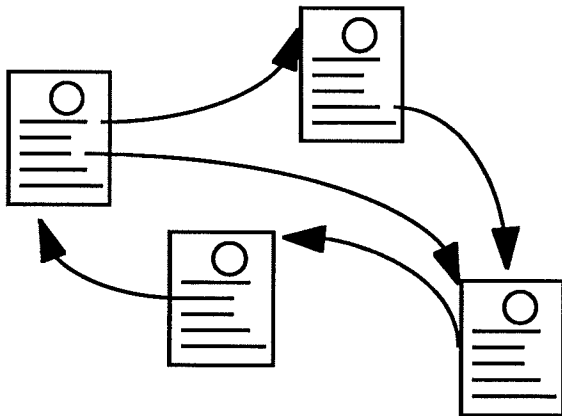


Figure X. In hypertext systems, documents are linked to one another.

After scanning the table of contents, most users will open a book in the middle and flip through the pages until some graphic, chapter title or block of text catches their eye. They may start reading from that point or turn back to the start of that chapter. Computers cannot directly reproduce this activity, but hypertext comes close to the notion of flipping through pages.

In hypertext, the author links words, phrases or graphics to other documents. For instance, a scientist might link a footnote citation to the actual source document. In another hypertext strategy, the author might link one part of the document to another, perhaps linking a passing reference early in the work to a later chapter that treats the subject in depth.

Hypertext systems that support many kinds of data, like the World Wide Web, can be quite rich environments as they allow users to link to audio, video, 3D models, animations, and other digital media.

Index: Searching

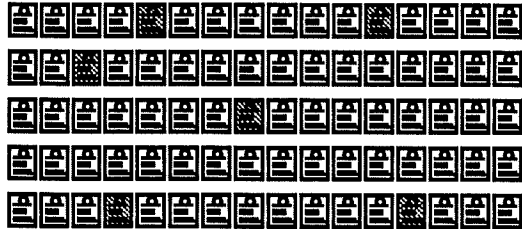


Figure X. A searching system can respond to a query by identifying documents that match query terms.

When readers want to find all the references to a specific subject, they turn to the index in the back of the book. This is akin to a searching interface, in which users can enter a query and receive back a list of the documents that contain information on the queried subject. On the Internet, the Wide Area Information Servers, or WAIS, system offers full-text searching of large collections of information. World Wide Web, Gopher and dedicated WAIS clients all offer searching interfaces to WAIS sources.

Updating. Staying up to date with our current interests can be difficult unless a steady stream of filtered information is automatically presented. If this is not done well enough, then we just won't find the time to read it. This process is similar to newspaper production, in which a staff of reporters and editors filter and interpret information, present it in an appealing way on a page, and the daily paper is delivered to subscribers' doorsteps.

Information Integration. Users will require seamless access to all information -- personal, corporate and published -- ideally with the same tools. This breadth of information will probably not be in one library or database but rather include one's own files, enterprise databases, and many outside sources. Searching must be easy and intuitive even through the mountains to search through are large and unorganized.

WAIS: The Server as Navigation Tool

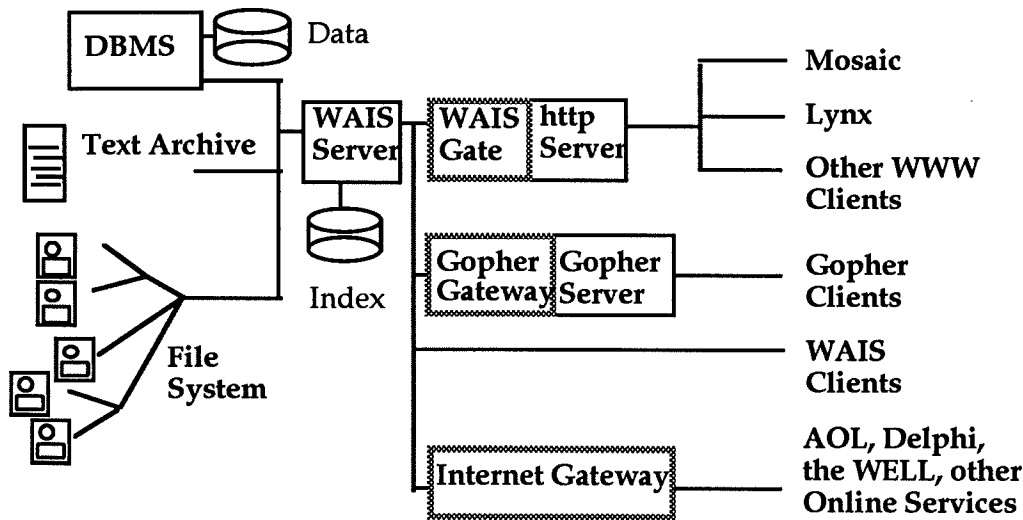


Figure X. WAIS servers can consists of several kinds of databases and can be accessed by a wide range of users, via gateways to other Internet servers and to commercial online services.

The server's role in helping users find the most relevant documents from potentially millions is to gather the information, organize it, and present it in meaningful ways. The server stores and indexes information and offers it to users through standard protocols.

Gathering Information

Whether the original documents are in external databases, newfeeds or file systems, the server must collect the documents to index them. If the documents can be used in their original formats and automatically converted, then expensive hand-editing costs can be avoided.

Organizing the Documents

Segmenting the data collection into meaningful sections can aid in searching and browsing. Sometimes documents have some structure that can be expressed through fields (such as electronic mail headers or SGML tags), metadata codes (such as on news feeds), or positions in a tree (such as a file system). A server can express this organization to users as an aid in navigation.

Browsing the collection is helpful in understanding the breadth of the collection. Different searching techniques such as natural language search, relevance feedback, and boolean searching aid in the process of winnowing through large collections without relying on the publishers catagories.

Presentation of the Information

The server should offer the documents to as many users as possible which can mean automatically reformatting them for different user interfaces. For instance, some interfaces support hypertext links, so creating these based on the structure of the documents can add

significant value. In this context, the World Wide Web server is a client since it controls the user experience.

Navigating a Sea of Servers: Ubiquitous Protocols

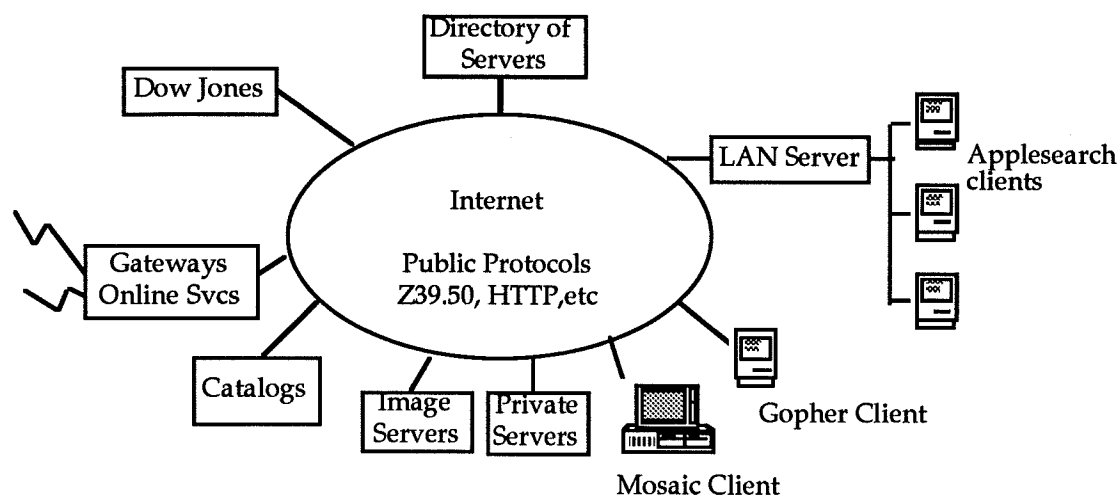


Figure X. Global networks are integrated by standard protocols.

While client-server has significant value within an organizational LAN, the real power is evident on wide-area networks like the Internet. When a good protocol is in place, network users can access multiple servers in a single search, talk to many different kinds of servers in the same way, access personal, organizational and published (wide-area) information in an integrated fashion, use sophisticated clients, and use the network as a reference source with such tools as a directory of servers.⁶

To provide all of this, the ideal protocol must be flexible, extensible, standard and, of course, expressive enough.

Flexible. The protocol should take advantage of the desktop personal computers as well as the server machines. Furthermore, it should allow for navigating of many data types -- not only text but also maps, DNA structures, and other unusual data types. Finally, it should allow clients to automate the gathering process.

Extensible. An extensible protocol can grow add new features without going through a long standardization process.

Standard. The protocol should be based on non-proprietary, international standards, so companies can compete but still be interoperable.

⁶ Protocol committees that are working on relevant network standards include: Internet Engineering Task Force, National Information Standards Organization, International Standards Organization, OSI Working Group on Library Applications. Document formats standards are set by other groups and companies.

Expressive. To be good enough, the protocol must be able to handle the current set of needs and be able to retrieve any kind of data, including text, graphics, sound and video.

The Future of Client-Server: Agent Technology

In the future, client-server technology will be exploited to develop agents that will serve as alter egos. Agents will be able to ponder indications of a user's preferences and act accordingly. A user's computer will know what its owner reads and doesn't read, to whom messages are sent, whose messages are ignored, etc.⁷

Automating some of the information collection tasks can help find relevant information from thousands to tens of thousands of sources. Given the power of desktop machines and a protocol that allows for machine automation we have the pieces needed to create these searching automatons. On the Internet, there are literally thousands of information servers, so a system of robots would be useful.

While the word "agents" suggests a human capability similar to a secretary or research assistant, the current technology is in its infancy. The precursors are present however: a growing body of quality information, computer-to-computer protocols that can support agents, multi-tasking operating systems on the desktop, digital networks and most importantly a discerning user population.

Today experimental agents are starting to perform the following tasks:

- Ask many servers a question on behalf of a user and track the user's actions in response to the answers.
- Scour the world (within a budget) to find new sources.
- Work 24 hours a day finding information.
- Format "personal newspapers" for users to read off-line on portable machines.
- Gossip with other clients to share information.

Organizations like Xerox's Palo Alto Research Center, Massachusetts Institute of Technology, General Magic and Ensemble are working to make these capabilities available in the next few years. By the time users start to use these automating processes, they may not even be aware of them.

⁷ Commercial systems include AppleSearch (developed from Apple's Rosebud project) from Apple Computer and Relevant Personal Digital Newspaper by Ensemble Inc.

Technology for a New Industry

While the Gopher, Web, and WAIS technologies have proven popular on the Internet they have not yet proven themselves in large-scale commercial publishing. At the time of this writing, a number of trials combining the strengths of the previously separate systems were underway. Many questions remain: How much will networked information cost? Is subscription or pay-per-view a better payment model? Is the current technology good enough to replace current niches of commercial publishing?

The premise that high-quality network publishing systems will result in a full-fledged information economy, in which there is payment for information, cannot be tested, until security and online transaction mechanisms are in place. Internet security is in its infancy, although the technical pieces are now in place.

Security in a World Wide Network

In the network publishing environment, security systems restrict access to documents based on the user's identity. Unlike services that "broadcast" files (like NetNews or CD-ROM distribution), in a network publishing system, documents are only copied when a user requests them. Thus the publisher can control the distribution of the work, simply by telling the server which users are allowed access to which documents. Even so, many potential publishers and users have serious concerns about privacy, theft and viruses.

Privacy. Network publishing brings up new issues of privacy because most users are unaware that their actions can be recorded. For instance, WAIS, Gopher and World Wide Web all generate usage logs for the server. These logs tell the server administrators who searched for what information and which files were downloaded. While these logs provide valuable information that helps information providers improve their services, they also represent potentially valuable information that could be used against a user or, less ominously, sold to third-party marketers, much as print magazines sell subscription lists.

Encryption can protect information during transmission by encoding messages so they can only be read by the intended recipient. But this is only part of the answer. The network publishing community also needs to develop rules of conduct for handling this information.⁸ Already some services are advertising that they do not sell or trade information in their logs.

Theft

In this context, theft refers primarily to improperly reselling information owned by someone else, or giving away copies of something that is being sold. Network publishing systems are attempting to make it easy for users to act legally by making "pointers" to the original data. Someone who wanted to include an article from a for-pay online magazine could simply construct a pointer that would bring the user to the site of the legitimate publisher of that document.

⁸ Brewster Kahle. 1991. "Ethics of Digital Librarianship." URL

Another concern is that someone who had not paid for access to a for-pay server would be able to break in, thus depriving the provider of income. This possibility is effectively handled by authentication procedures.

Viruses and Security Breaches

In the WAIS system, users do not actually log in to the server; rather, they search through a read-only application layer protocol (Z39.50⁹). Thus there is no risk of information on the server being modified or of the server being contaminated by viruses.

Current Security Techniques

Securing information involves a balance of ease-of-use and protection. It would be prohibitively difficult for a user to remember different passwords for every server contacted; on the other hand, giving a user a single password for many information servers would invite abuses.

Hardware encryption devices, like the proposed Clipper chip, are far more difficult to distribute than a software solution. Older security systems that allow two systems to communicate because they each know the same secret key¹⁰ are difficult to extend to a system with thousands of servers and millions of users. Two new systems have been developed to address these problems -- public key and Kerberos -- and both are starting to be used in the WAIS environment.

Public key technology offers all the right pieces: privacy, scalability, authentication and digital signatures. The catch is that it requires licensing from a private company.¹¹ This has not stopped implementation but it has slowed dissemination. Kerberos, from MIT, does not require licensing, but requires a hierarchy of authorities to validate connections.

All in all, the public key and Kerberos technologies offer strong security measures for network publishing, but the dissemination of the infrastructure will take awhile.

Billing and Payment models

Making it possible for small producers, as well as large, to be compensated for their work offers opportunity for continued growth in the Internet. Every company, every academic department, every family should be able to publish on this network. Some will want to be compensated. Facilitating this future industry is one of the goals of the WAIS system.

⁹ The WAIS protocol suite is described in "WAIS Profile of Z39.50-V2," OIW SSIGLA.
<ftp://ftp.wais.com/WAIS-PROFILE-of-Z3950-V2.txt>

¹⁰ Data Encryption Standard, originally developed by IBM, is a U.S. government standard. It is described in National Institute of Standards and Technology FIPS Publication 46-1, Jan. 22, 1988.

¹¹ RSA Data Security Inc., Redwood City, CA

Collecting the customer usage information is technically not difficult, but many questions about how the business will develop remain unanswered. There are many possible billing structures: subscription, site-license subscriptions, pay-per-article, advertising-supported, and many others. While print publishing broke up into separate industries -- writing, publishing, printing, distributing, and retailing-- the network publishing business might evolve differently. In the current stage, the goal is to reach a critical mass of quality services that readers are willing to pay for.

Current businesses that are making money on the Internet include connectivity providers, hardware providers, telecommunications companies, book publishers and consultants. After the plumbing is done, then interactive information services such as magazines, games and performance events can proceed on the networks.

Some Internet information service providers (such as WAIS Inc., Bunyip and Pandora) are just starting to make money. These businesses typically take content from a publisher and re-target it for the networks. Some publishers (such as Encyclopaedia Britannica) operate the systems themselves, but niche service bureaus offer expertise and economies of scale.

How the customer will pay for information access is another open question. End-users might be billed for single subscriptions, but more likely connectivity providers (such as regional Internet providers and online services) will act as middle-men to centralize billing.

Conclusion

Weaving the network publishing elements together to make a usable system is the goal of the WAIS system. By incorporating a large number of servers and users, an open protocol for future growth and compensation mechanisms, such a system can grow. So far the system has been useful on the Internet for search and retrieval, and WAIS resources have been blended into many other systems such as Gopher, World Wide Web, e-mail services and others. To date over 100,000 users have used WAIS and the number continue to grow.

Network publishing is not about saving trees or replacing books; it is about new relationships between publishers and readers, a fundamental shift in the way people obtain information, and new forms of literature that will spring from a people unleashed to create and publish in an inexpensive new medium.

This book first describes the client, server, and protocol technologies that are used in accessing WAIS servers. Then representative applications are described as example uses of the technology. Finally, some of the trends in the technology are extrapolated into a predictions on where this new industry is headed.

CLIENTS: EXPLORERS, SEARCHERS, AGENTS

The user's program for cruising the networks is called the "client" program. Within the context of network publishing, a client is a navigation tool -- user software that provides the ability to

navigate through large amounts of information. People using the networks to gather information engage in three modes of navigation:

- browsing, or the ability to explore available resources;
- searching, or the ability to get immediate answers;
- agenting, or the ability to have new information delivered automatically.

The dominant client on the Internet is Mosaic. (Originally developed at the National Center for Supercomputing Applications, there are now many commercial and freeware versions of the program. Similar programs, such as MacWeb from EInet, are also available.) While Mosaic is primarily used as a client for the World Wide Web system (described in more detail below), it actually serves as an integrated Internet client, capable of contacting other services -- including WAIS, ftp, gopher, etc. -- and presenting the results in hypertext format.

The Web is a browsing system that uses hypertext links to point users to related documents on the same server or to another server half way around the world. When the user clicks on hypertext, usually indicated with underlined or colored text, Mosaic contacts the server where the linked document is located and asks for the document, which is indicated by a URL, or Universal Resource Locator. The server responds to the request by sending the document to the user's computer. Mosaic then interprets and displays the document. In this way, users can "surf" the Web, following links that interest them, browsing through information.



Welcome to MGH Neurology!

This Webserver is maintained by the Department of Neurology at Massachusetts General Hospital, under the direction of the Chief of Neurology, Dr. Anne Young.

Click on items to explore!

Computer services and networking provided by *John Lester - (lester@helix.mgh.harvard.edu)*

First, please try out these links to help you learn about this powerful new technology and how to use it efficiently...

- Find out the latest news concerning this Webserver by clicking here.
- Please click here for a tutorial of how to navigate the Web.
- Learn about the "World Wide Web Initiative" Project.

Figure 2-1. Massachusetts General Hospital Department of Neurology's World Wide Web server includes underlined hypertext links that let users learn more about the department.

The WAIS system, on the other hand, allows users to search very large collections of information (hundreds of gigabytes) to find specific documents. While some of the functionality of the WAIS system is currently only accessible through dedicated WAIS clients (to be discussed in more detail below), searching can WAIS can also be performed within Mosaic. Many Internet services have integrated Web and WAIS servers, so that users can both browse via hypertext links or search for specific documents.

The WAIS system allow users to ask "natural language" queries, that is queries with no special syntax or characters, to search the full-text of all documents in an information source. Some WAIS servers offer a variety of more sophisticated searching techniques.

Information agents, like AppleSearch and Relevant, build automation into both searching and browsing. In their current state, they go out to certain servers and look for new information about topics in which the user has expressed interest. Ultimately, they may do much more -- such as evaluating competitive information, completing financial transactions and more.

WAIS in Business: A Scenario

To give an overview on how a professional might use the Internet in business, the following scenario shows how an executive might use these navigations techniques to find a lead, research a company, develop background on an industry, and get enough information to make a personal contact. While this story is fictitious, the servers referred to are currently available on the Internet.

Every morning, when Christine Anderson, vice president of sales at Telephony Consultants, comes to work, she turns on her computer, connects to the Internet, launches Mosaic, and checks the morning's electronic edition of The Wall Street Journal. The newspaper is part of the DowVision information service by Dow Jones Inc. Even though the Internet version of DowVision is located on a server in the San Francisco Bay Area and Christine is in Pittsburgh, she can search interactively, getting immediate responses to her queries.



Figure 2-2. Top news stories in The Wall Street Journal as presented in Mosaic. While Mosaic is a Web browser, DowVision on the Internet is actually a collection of databases indexed on a WAIS server. (The server receives an automatic newsfeed, periodically indexes the documents and makes them available on the Internet.) By using a gateway between the WAIS protocol and the Web protocol, WAIS searches can be packaged as hypertext documents and viewed on Web browsers. This is how Christine views her morning paper.

Always on the lookout for new business, Christine takes a look at the Prexxtron story to see where the company is relocating. To read the whole story, she simply clicks on the headline and the story is displayed.

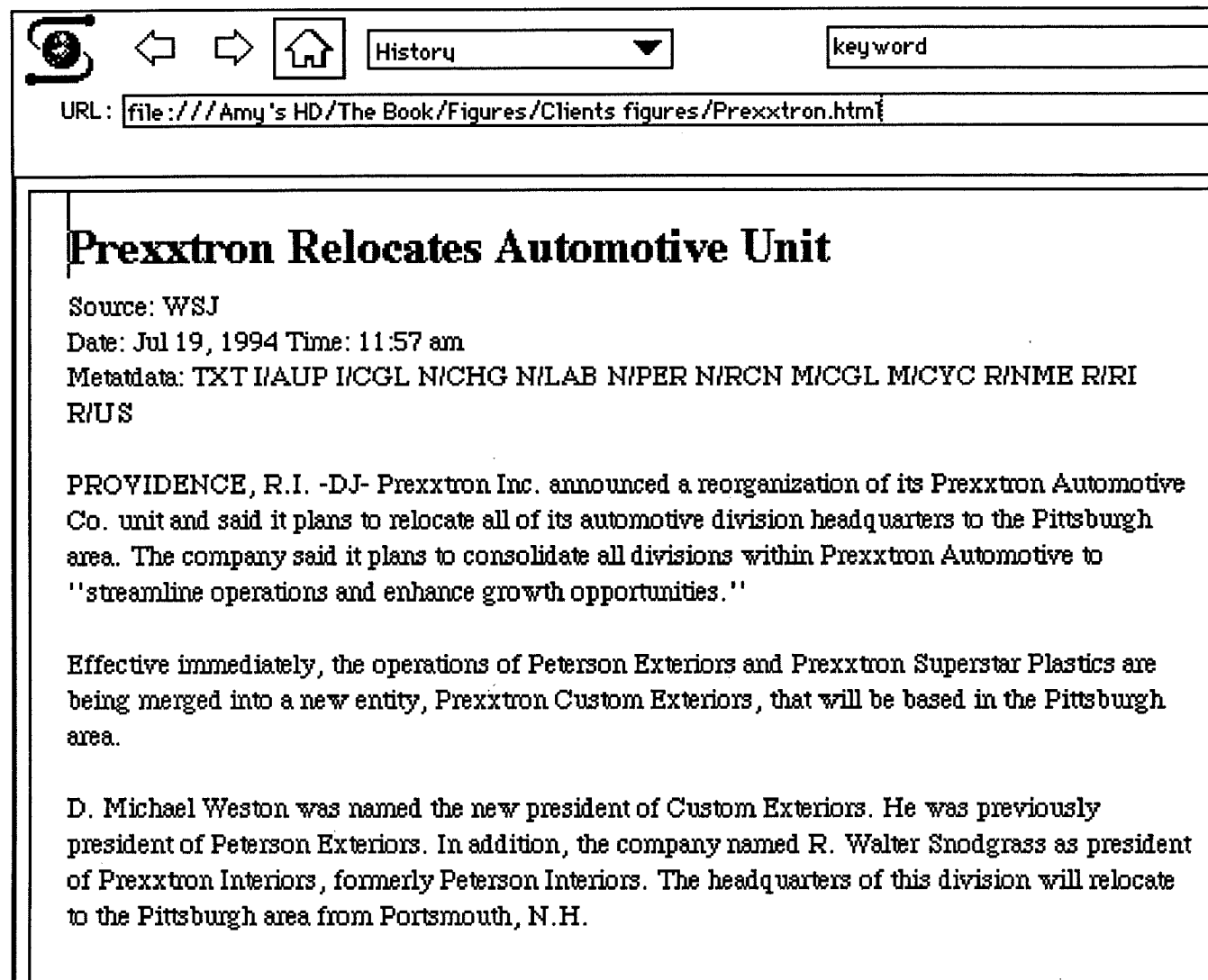


Figure 2-3. About two seconds after Christine clicked on the headline, the article is displayed on her screen. By clicking on the headline, Christine told the WAIS server to show her the document. The server located the document, packaged it as a Web document and sent it to Christine's computer. Mosaic then displayed the article on her screen.

"Hmmm," Christine says to herself, as she reads the article. "They're moving to Pittsburgh. I have to talk to them about their telephone systems. I'd better learn something about this company." To find more articles about Prexxtron, Christine searches all the DowVision wires with "Show me more about Prexxtron Corp." About two seconds later, she gets back a hit list of headlines to documents that match her request, ranked in order of relevance. Each headline is a hypertext link. The hit list is shown in Figure 4.

***Picture of searching dowvision with query "prexxtron corp."

Figure 2-3. When Christine presses the "Send" button, she is sending a query to the WAIS server, asking to search all the DowVision databases. The server doesn't actually understand the sense of the question; instead

it takes the uncommon words in Christine's query ("Prexxtron Corp.") and searches an index of all the words in all the databases. The entry for Prexxtron in the index points to the documents that contain the word. The server then gets the headlines for those documents, creates a hitlist, packages it up as an HTML document, and returns the hit list to Christine.

***Picture of hit list

Figure 2-4. The hit list is relevance-ranked, with the most relevant documents at the top of the list. Each headline also has a score on a scale of 1 to 1000. The top story always has a score of 1000.

Several stories of the headlines look good to Christine. To receive the ones she wants, she clicks on the headline and a few seconds later the actual document is displayed on her screen.

In the course of her research, Christine has determined that Prexxtron may be a strong sales prospect. They are moving a factory and executive offices to Pittsburgh. In addition, they are part of a geographically distributed company, so they may be interested in Internet connections. Christine asks Tom Stevens to put together a preliminary proposal and presentation for Christine to use in her initial meeting with Prexxtron.

Tom searches Telephony Consultants' current and previous contracts to see if they have already done a job similar to Christine's description of Prexxtron. For his more focused searching, Tom will use a dedicated WAIS client instead of Mosaic. He uses WAIS for the Macintosh, an early client that features click-and-drag operations. (put sidebar about searchers interface opposite this page).

Because Telephony Consultants has many offices around the country, each with its own set of databases, the company uses the Internet to connect all the offices together. Only Telephony employees are allowed to access these databases.

To get a list of all the contracts databases in the company, Tom first contacts Telephony's "directory of servers," a database of all WAIS servers in the company. There is also a directory of servers for publicly available databases and, of course, many other organizations run their own directory of servers.

The WAIS for Macintosh program, shown in Figure 5, has several different windows. The top window is where queries are entered. The middle right window contains a list of sources to be searched. The bottom window displays search results.

Since Tom's client is configured to always start with the Telephony directory of servers in the Sources window, Tom just enters his query: "Show me all the contracts databases." He clicks on the "Run" button and a few seconds later he gets a list of all the company servers that have something to do with contracts.

**Figure 5- above query with results

Figure 5: Searching the directory of servers yields a hitlist of source descriptions.

Tom can get more information about any of these sources by opening up their source description files, which contain descriptions of their content, as well as other technical information. The directory of servers is actually a database of source description files for all the company servers. When Tom searched for "contracts databases," the server checked the descriptions for those words and returned those descriptions that matched.

****Figure 6: Source description file**

Figure 6: WAIS source description file

Tom wants to search all the contract databases, so he drags all the source icons from the Results window to the Sources window. Then he enters his query and clicks on Run. Tom doesn't enter just a natural language query, though. He wants to search very specifically for a contract that matches the Prexxtron specs, so he uses a combination of Boolean and fielded searching. These are functions available in advanced WAIS servers.

To start with, Tom knows that the contracts are structured with entry fields for different kinds of information. For instance, there's a field for type of business, site specifics, kind of line, number of handsets, and so on. Tom wants to search for contracts for factory installations at manufacturing sites, so he types "site=factory AND business=manufacturing OR mfg."

The words AND and OR are Boolean operators; they require certain conditions to be met in order for a document to match the query. AND means both conditions must be met, OR means either condition may be met. In this case, Tom is searching for contracts that contain the words "factory" in the site field *and* "manufacturing" *or* "mfg" in the business field. Then to narrow things down even more, he adds a search for "Internet" in the requirements field.

****query screen shot**

When he hits send, his query is sent to each of the contract servers in his Sources window. They each check their databases for files that match Tom's query. His hit list reflects matches from all of the servers and is ranked by relevance.

Tom checks out the hits, viewing the contracts that look appropriate, and soon he finds one that looks quite close. The next step is to find the initial presentation made to this client, so he can use it as a template for the Prexxtron proposal.

Since this client was based in Durham, N.C., Tom will just search the databases at Telephony's southern office in Atlanta. He searches the company directory of servers for "Atlanta AND presentations." There is only one database that meets both criteria and Tom searches it for the client name.

Sure enough, there it is in the hit list, which also shows that it is an Aldus Persuasion document. When Tom retrieves the actual file, Persuasion is launched on his machine and the file displays. Now Tom quickly edits the file, inserting Prexxtron's name and changing the specifics of the presentation. He sends the new file to Christine via email.

Christine looks at the presentation, makes a call to Prexxtron, tells them what she has in mind for their phone needs. Prexxtron is so impressed they set up a meeting immediately. Now it's up to Christine to land the deal.

Searcher's Interfaces (Sidebar)

The WAIS for Mac client that Tom used in the story above was derived from the original WAIS client called WAISStation. It was designed for senior partners at a large accounting firm and it had to be useful within 30 minutes to provide answers quickly on focused topics. The client was developed with three goals in mind:

- Provide access to personal, corporate, and published information from one interface,
- Use simple searching and browsing constructs -- "No algebra!"
- Take interactive questions and automating them into an agent

While not designed for the net-surfer, it still had to be fun as well as useful. The resulting interface used several techniques to achieve these goals:

- Multiple servers can be searched at once whether these servers physically distant
- A directory of servers was used to locate new server offerings
- Commercial databases were used that required restricted access and payments
- Drag and drop was used to indicate a particular document was relevant
- Saved queries can be automatically re-asked to keep the user up-to-date.

Similar searching clients have been developed for most computer platforms. This section gives a quick review of some of them.

Agents and Personal Newspapers

As more and more information becomes available online, it may not be enough to be able to navigate it through applications like WAIS and the World Wide Web. The next step is information agents, that do the work of sifting through documents and alerting users to new and relevant information.

"It's simple," writes Nicholas Negroponte of MIT's Media Lab¹². "Just because more bandwidth exists, don't squirt more bits at me. What I really need is intelligence in the network and in my receiver to filter and extract relevant information from a body of information that is orders of magnitude larger than anything I can digest. . . . Imagine a future

¹²"Less Is More: Interface Agents as Digital Buffers," *Wired* 2.06, June 1994.

where your interface agent can read every newspaper and catch every broadcast on the planet, and then, from this, construct a personalized summary."

An agent-based system, according to Negroponte, is "not a matter of a questionnaire or a fixed profile. Agents must learn and develop over time. . . . It is not only the acquisition of a model of you; it is using it in context."

These information agents of the future would be intelligent enough to discern user interests from previous activity, responsive enough to update their understanding of the user's interests, flexible enough to accommodate direct searches as well as automatic searches, and transparent enough so that normal work would not be interrupted.

Needless to say, commercially available agent-based programs are not quite up to this task. With programs like AppleSearch and Relevant, we have simple agents, the sort that can work from a fixed profile.

AppleSearch

AppleSearch is a client/server software product that was developed from Apple Computer's work on Rosebud¹³ during the initial WAIS research project. AppleSearch is a LAN-oriented product, in which users on the LAN can perform full-text searches of an unstructured database residing on a server on the LAN. In the original version, AppleSearch only supported a proprietary protocol, but version 2 (which was to be released in the second half of 1994) also supports the Z39.50 protocol. This means that AppleSearch sites have access to WAIS database on the Internet.

Agenting in AppleSearch takes the form of "reporters," search agents that can be scheduled to search one or more servers for specific information. For instance, a user might create a reporter to search for news about Microsoft Corporation on DowVision. This reporter could be scheduled for 9 AM and 5 PM every day.

****Figure 14. Picture here of a reporter result****

These automatically updating reporters are a first step to personalized newspapers, but AppleSearch doesn't take the next step of actually combining reporters from different sources into a single unified personal daily (or hourly) newspaper.

Strictly speaking, AppleSearch is not a client/server application but rather a client/client/server application. This is because the LAN server acts as the client in the transaction with the remote server. The LAN server then passes responses along to the end user's computer. Similarly, searches (in the form of reporters), are not made directly by the end user but are passed along to the LAN server. Searches are actually triggered when the server automatically indexes updated databases and results are passed along to the user at the time the reporter is scheduled.

¹³ Cite the Clients paper from 89

This client/client/server architecture allows AppleSearch sites to provide "managed access" to information. Since users have to go through the local server, system administrators can completely control which remote servers will be available.

ess" to information. Since users have to go through the local server, system administrators can completely control which remote servers will be available.

Ensemble Personal Digital Newspaper

Ensemble Information Systems' Relevant software does deliver a personalized newspaper, specifically The Wall Street Journal and The New York Times.

Relevant is a front end to Dow Jones' DowVision information system (the same service illustrated at the beginning of the chapter), which carries both newspapers. Unlike Mosaic, however, Relevant is not a browser. Instead it's a program for Internet users who don't want to spend a lot of time poking around for relevant stories. What Relevant provides users is a way to get an electronic version of the daily paper, formatted and online, and to search for stories by selecting metacodes.

Users typically download each day's edition of the Wall Street Journal or New York Times (usually via email, but TCP is also supported). The interface to the daily paper is divided into two sections. On the left is a listing of all the departments in each section of the paper; on the right is a list of headlines for all articles in the selected section. When the user clicks on a section heading, a new list of headlines is displayed on the right-hand window. When the user clicks on a headline, the actual article is displayed in a separate window.

THE WALL STREET JOURNAL via DowVision

© 1994 Dow Jones & Company. All Rights Reserved

Monday, June 27, 1994

From Ensemble Info

<div data-bbox="152 317 228 380"></div> <div data-bbox="261 327 477 359">Front Section</div> <div data-bbox="167 396 496 636"> <p>Economy International Leisure & Arts Editorials Letters to the Editor Politics & Policy All Front Section Stories</p> </div>	<div data-bbox="552 317 628 380"></div> <div data-bbox="638 327 914 359">Money & Investing</div> <div data-bbox="571 396 860 690"> <p>Absorbed of the Market Dividend News Foreign Exchange Heard on the Street New Securities Issues World Markets Your Money Letters All Money & Investing</p> </div>	<div data-bbox="984 317 1357 348">Page One -- Front Section</div> <div data-bbox="971 369 1542 474"> <p>'I Loved Women And Horses; I'm Sad I Die' 'Merry Cemetery' of Romania Is Proud of 'Wife, Bring Me Lunch'</p> </div> <div data-bbox="967 501 1544 642"> <p>Storm at Sea: As Salmon Catch Falls, U.S. a Into a Heated Dispute --- Both Nations Cla Which Range Vast Areas Of the Northern 'Transit Fee' Riles Americans</p> </div> <div data-bbox="967 672 1544 711"> <p>The Outlook: Health-Care Inaction Can Carry</p> </div> <div data-bbox="967 739 1544 879"> <p>The Unloved Buck: Dollar's Fall Threatens U Makes Global Markets Dicey --- A 17-Nat Prop Currency Fails, Causing Jitters on Int New Snag in Japan Trade</p> </div> <div data-bbox="967 909 1456 942"> <p>What's News -- Business and Finance</p> </div> <div data-bbox="967 972 1343 1005"> <p>What's News -- World-Wide</p> </div>
<div data-bbox="152 674 228 737"></div> <div data-bbox="277 684 470 716">Marketplace</div> <div data-bbox="167 751 482 1056"> <p>Business Briefs Corporate Focus Education Enterprise Law Marketing & Media Technology Who's News All Marketplace Stories</p> </div>	<div data-bbox="552 779 628 842"></div> <div data-bbox="667 789 899 821">Personal Index</div> <div data-bbox="552 884 628 926"></div> <div data-bbox="638 884 876 915">Special Section</div> <div data-bbox="651 947 837 978">Corrections</div> <div data-bbox="565 1010 922 1041">DowVision from Dow Jones™</div>	

Figure 15. When a section of the Wall Street Journal is selected, the window at the right shows headlines for articles in that section. Here headlines for Front Section stories are displayed.

The New York Times News Serv

© 1994 The New York Times. All Rights Reserved.

Monday, June 27, 1994

From Ensemble :







 General News National International Editorials Politics Obituaries Weather	 Living Arts Arts & Entertainment Lifestyles Reviews Science & Technology Sports	National ARE BROADCASTERS NEGLECTING EDUCATIONAL TELEVISION? BLACK CAUCUS FAULTS ADMINISTF AFRICA CONFERENCE BRADLEY TAKES SHOT ON HEALTH CONSOLIDATION IS FORCING CHANG INDUSTRY DOLLAR PLUNGES TO RECORD LOW YEN HOLLAND TUNNEL PICKED AS NATI HISTORIC LANDMARK IN DEALING WITH CONGRESS, CLIN DOWN DETAILS ON HEALTH STRA INDICATIONS OF COMPETING LEGAL
 Business Day General Business Corporate Financial Industry	 News Briefs  Personal Index  All NYT Stories DowVision from Dow Jones™	

Figure 16. Here the user is looking at headlines for the News Briefs section of The New York Times.

Relevant users can do more than just browse the day's paper, however. They can also search for articles or create personalized newspapers. Both of these tasks use DowVision's metadata -- a system of codes that are linked to every document on the system. Relevant translates these codes into meaningful phrases and lets the user search on them. When creating a personal newspaper, the user can specify certain topics to use in the paper. Relevant takes the codes for those topics and sends a query to the DowVision WAIS server. The response goes to the Relevant server, which organizes it in newspaper format and delivers it to the reader.

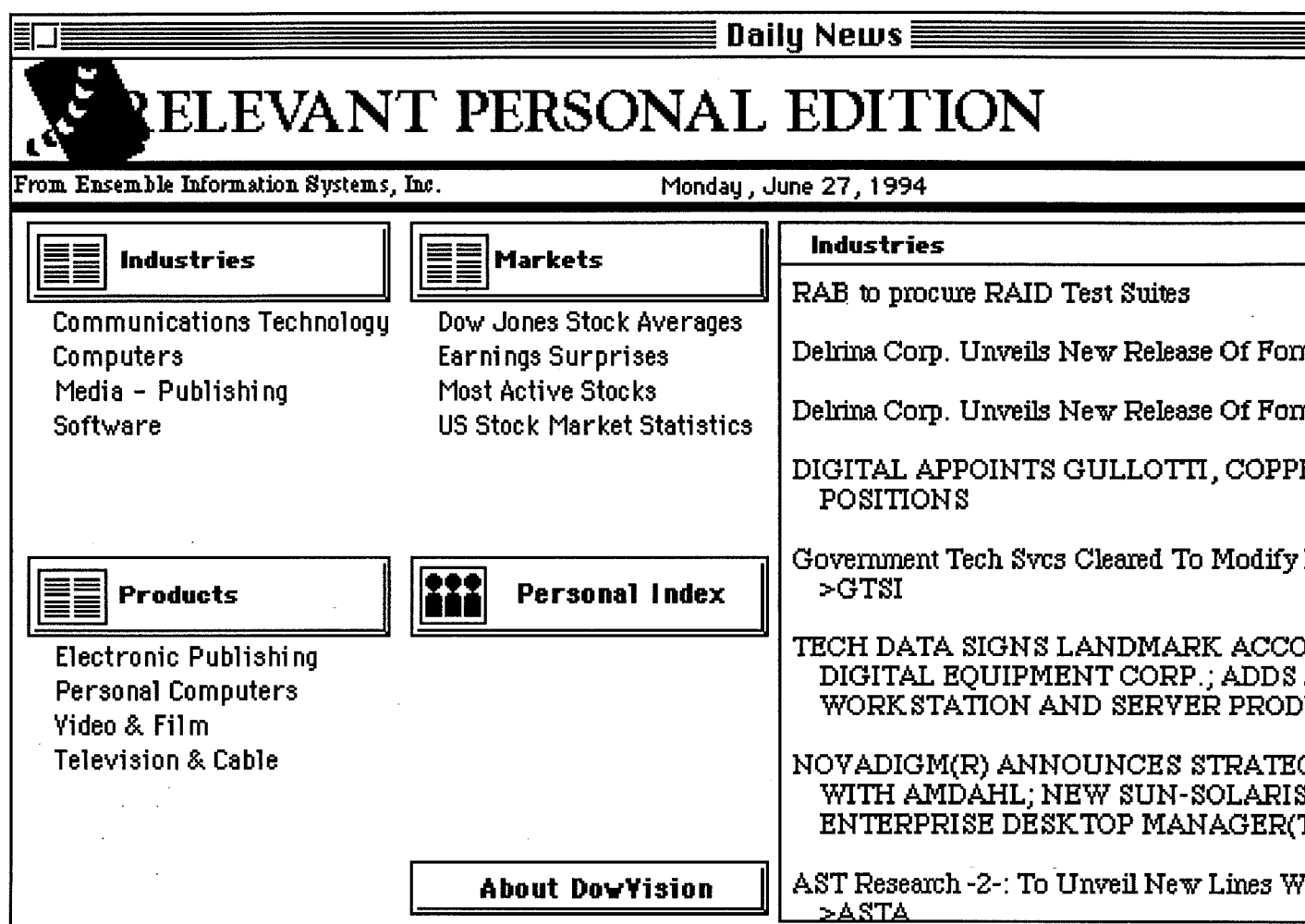


Figure 17. The Personal Digital Newspaper presents user-specified categories in the left-hand window and a hit list of documents within a category on the right-hand side.

The next step is agenting. Relevant can use these preformatted queries to go out every day and create a personalized newspaper of the latest news on these subjects, in addition to delivering the highlights.

Agenting in Macintosh clients

The publicly available Mac WAISStation client support a primitive form of "dynamic folders," a construct in which saved questions can be automated to search for updated information on a regular basis. This is an inflexible routine, however; the questions will search only the specific databases requested by the user when the question was saved. In addition, this feature users and appears to be scarcely used.

Minerva by Booz Allen Hamilton

Specialty interfaces are being introduced by a number of vendors for particular user communities. One is Minerva by Booz Allen Hamilton which is designed to make many different database systems available to users through a common interface.

Gateways

The Internet is becoming the backbone of the network publishing since it offers access to many user communities through gateways. Publishing on each online service is expensive because of the reformatting required for each. The ideal of the Internet is to publish work once yet have it me available from many systems -- while maintaining control of presentation, pricing and distribution. This is possible with WAIS because the protocol operates at a high enough level that data is represented in an abstract way. Gateways can then transform that information into multiple representations based on the client program being used. It is for this reason that Mosaic, Gopher and America Online users can also access the same information.

Gateways between the WAIS and World Wide Web systems were discussed earlier in this chapter. In this section, we take a relatively brief look at some of the other gateways such as America Online, Gopher and email. From this survey, it should not be hard to imagine gateways for such systems as Lotus Notes, Microsoft Exchange, or General Magic's Telescript.

America Online

In 1994 America Online (AOL) launched several Internet services available through the AOL interface. Under the Internet Center area, users can access both Gopher and WAIS servers, as well as NetNews, Internet mailing lists and email. This constitutes a major breakthrough in bringing the world of the Internet to the desktops of everyday computer users. While America Online does not currently offer the integration of graphics and text, as in Mosaic, it does offer something more important: easy-to-use access to the Internet for low-end users with slow modems.

AOL's Internet Databases option introduces a top level that divides the world into about 40 subject areas, from Art to Zoology. Inside these folders, the user finds a combination of both Gopher and WAIS sources. The Gopher servers are represented by folders, while searchable servers (both WAIS and Veronica) are represented by a search icon. Clicking on a search icon brings up an interface similar to the WAIS search document in Gopher.

The query field provides the same instructions that AOL offers for other search features in the system: "Type in words that describe what you're looking for, such as Boston and Washington." While this description is vague about how the search engine works, the interface, like other WAIS interfaces, is dependent on the capabilities of the server. If users knows the server's capabilities, they can write advanced queries accordingly, such as using Boolean operators in all capitals.

The AOL interface does not support relevance feedback and only allows for single-database searches. While the response list is ranked, it does not include ranking scores, size of file or name of database.

Also noticeable for its absence is the Directory of Servers, which was deemed by the developers to be too cumbersome and not effective enough in discovering servers. Instead they have opted for a more labor-intensive process of identifying servers by subject, renaming them and making them accessible through subject folders, as in Gopher. While this is a much more controlled philosophy than one typically finds on the Internet, it may be appropriate for an

online service that makes money for providing organization and ease of use to its customers. Indeed much of the Internet's complexity is concealed from users; for instance, when a server finds no documents that match a search, AOL masks the database description and "catalog" responses, simply showing an empty response window.

Certainly, other major online services like CompuServe, Prodigy and GENie are likely to offer their own Internet gateways and there will likely be a balance struck between full functionality and non-threatening ease-of-use.

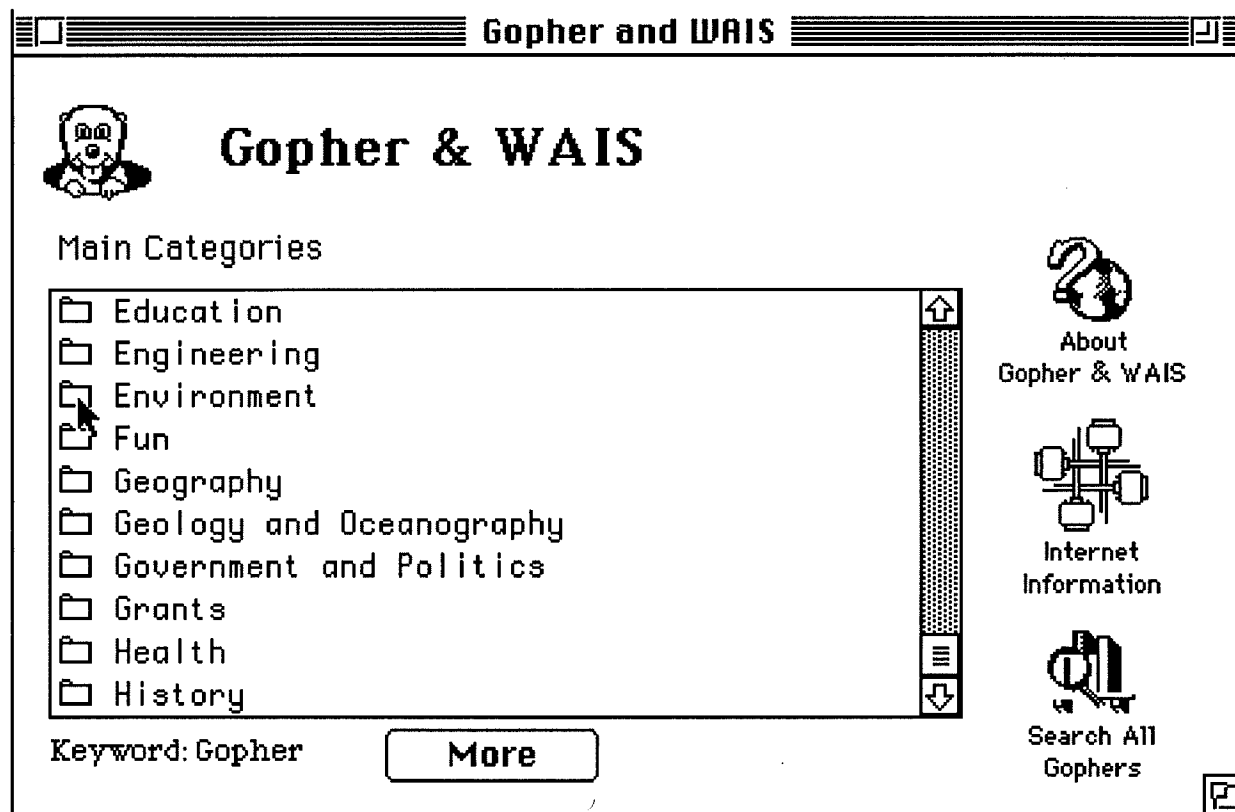


Figure X: America OnLine gateway to Gopher and WAIS databases on the Internet.

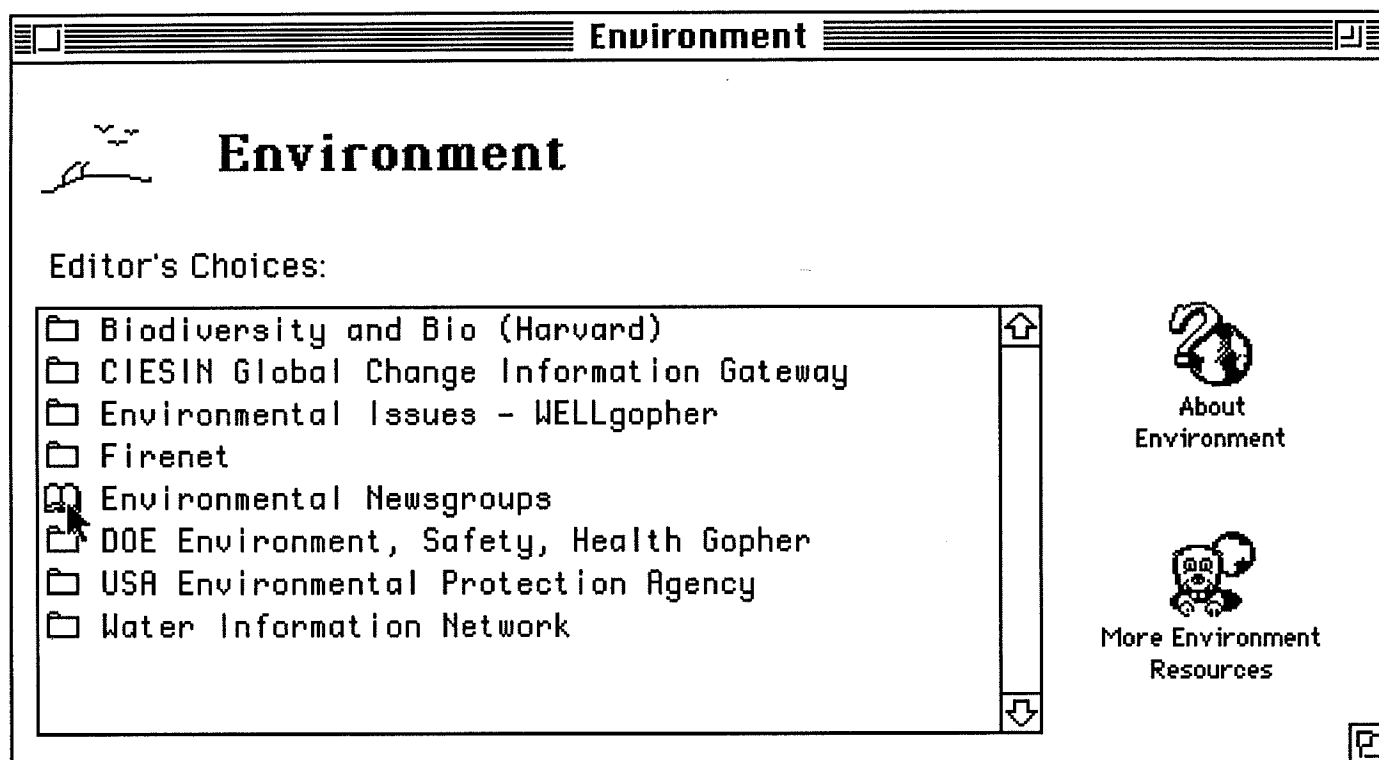


Figure X: The bookmark icon differentiates WAIS databases from Gopher folders.

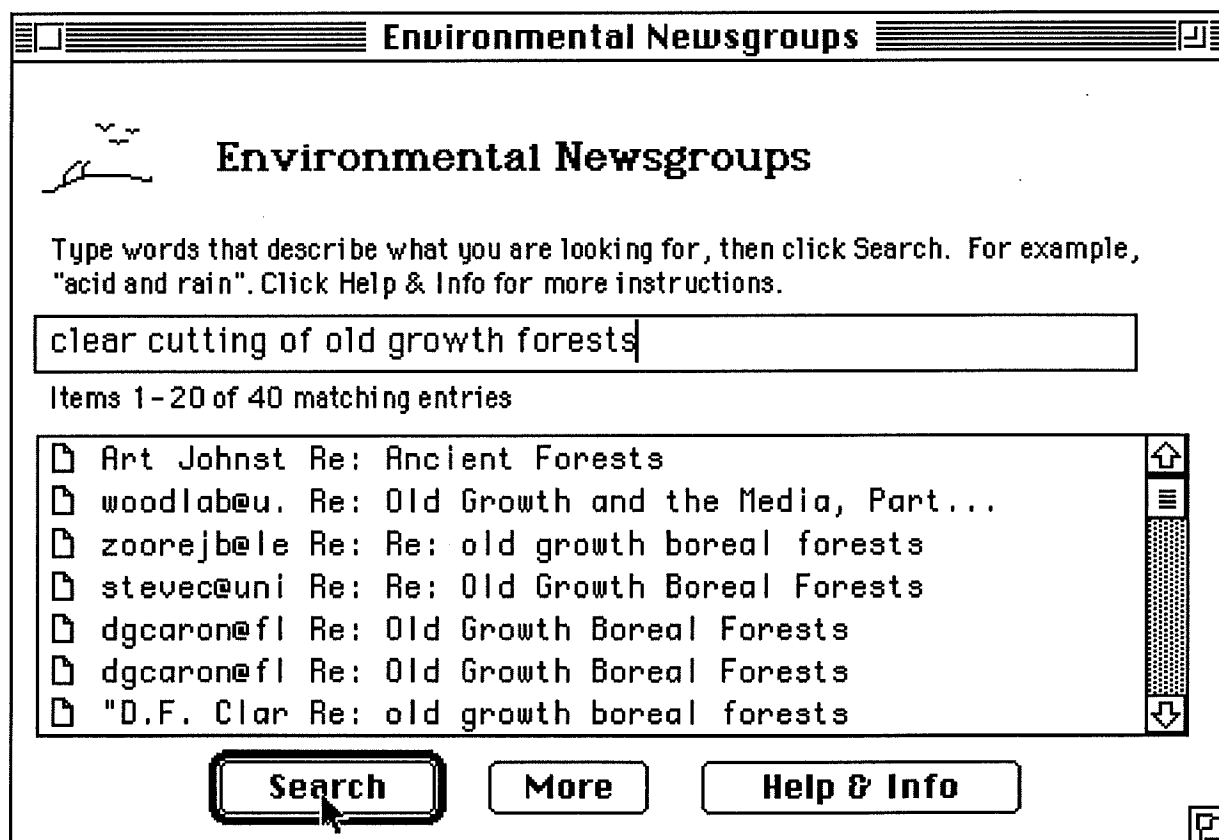


Figure X: A natural language search of a remote database.

Gopher

A system that uses a directories of files metaphor for global navigation is the University of Minnesota's Gopher. There are several thousand Gophers available around the world. By organizing information into a hierarchical structure, users can access information by moving through a series of menus and submenus, each of which brings the user to a deeper level, where there are more menus, actual files and access to searching capabilities.

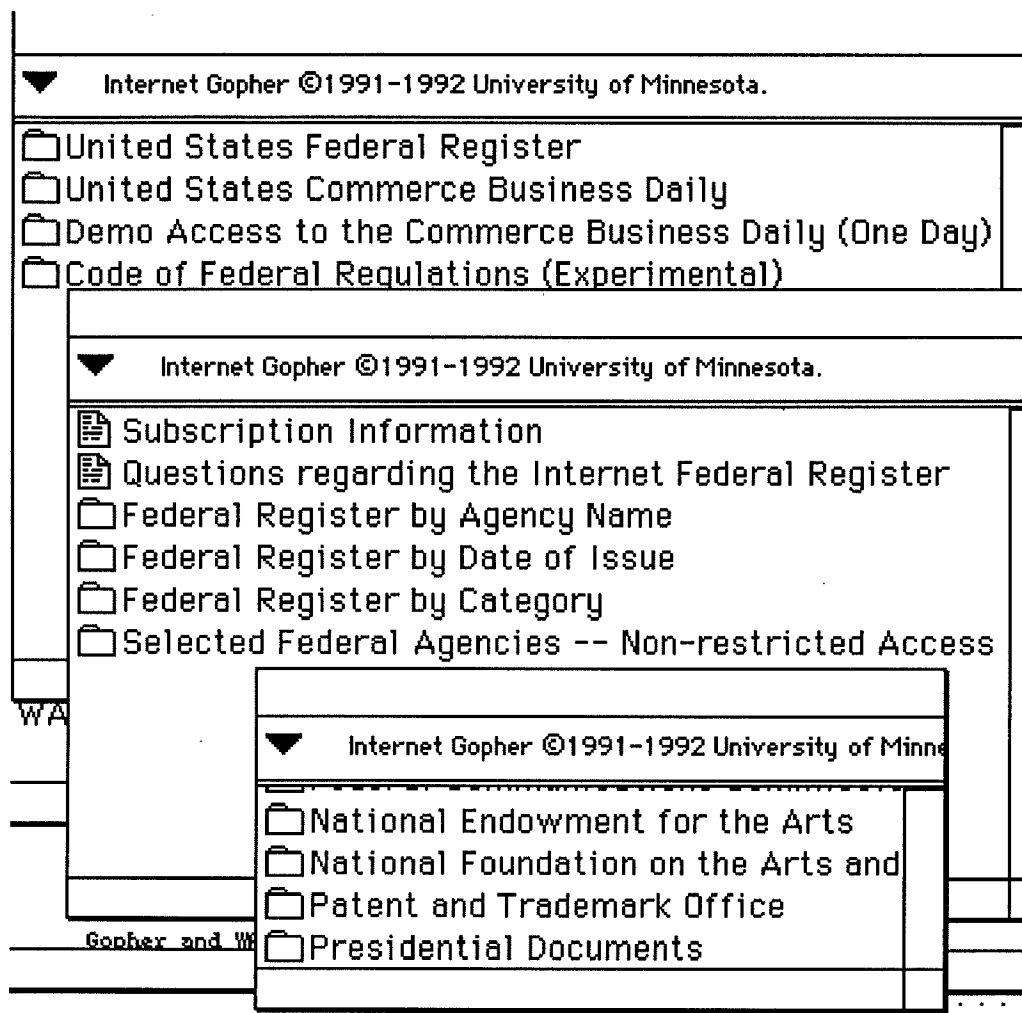


Figure 19. Three levels of Counterpoint Publishing's Gopher, using TurboGopher for the Macintosh as a client.

In Figure 19, above, we see three levels of a Gopher. This top level gives four menu choices. In this case, the user chose Federal Register and the second level offers two text files as well as several more menu items. Choosing one of the menus brings up a third level with more menu items -- a listing of federal agencies. There could be even more specific levels of menus. In the fourth level, for instance, there might be menus for the different departments within the selected agency. Eventually the user will come to some actual information.

add gopher with search box picture

While some Gopher servers provide only keyword searching, many actually use a gateway to the WAIS system. Theoretically, this provides the best of both worlds -- browsing and searching in one interface. In practice, however, the WAIS searching capabilities in Gopher are quite limited. Only one database can be searched at a time, the number of responses is limited and relevance feedback is not supported. In addition, security is problematic. Relevance feedback is most useful on databases over 100 MB but it is an important tool, especially for browsing.

The limited number of responses was not an issue in Version 1 of the Z39.50 protocol, since there was a limit of 50 to 100 responses. In releases of WAIS server software that incorporate Z39.50-v2, however, a potentially unlimited number of hits can be retrieved. This feature will require changes in the Gopher interfaces that may not be easy to implement.

Implementing security procedures for users of Gopher interfaces may also be difficult since requests are often forwarded through many machines. Therefore, the IP address of the original requester is often not available at the server side. This can be addressed by having the user connect directly from the users machine to the publisher's service when IP identity is needed. Another approach is to have the client always open a new Gopher server rather than forwarding through multiple gophers, but this is not standard behavior for Gopher interfaces.

Email Gateway

A public domain Email interface has proven to be a popular service on the Internet offering WAIS access to those that only have an email gateway. This interface can be tested by sending a message to waismail@wais.com and help messages will be returned.

Stanford University has been experimenting with a WAIS-based service based on netnews and technical report archives that offers periodic updates based on saved profiles. ***add email address for this***

A more advanced version might offer database filtering based on user name, forwarding to human attendants if a incorrect request was submitted, advanced logging, first sections of documents returned from a search rather than just headlines, etc.

While ASCII-based email is not a very rich interface, it does have the advantage that many more users can receive email than can use direct Internet resources.

GNU/EMacs

The WAIS interface for GNU-EMacs/Unix (GWAIS) is significantly different from the Macintosh interface (or any of the graphical interfaces, for that matter). It was developed specifically for use on the Internet by technically advanced Unix users. The design of the interface is a cross between WAISStation and other EMacs interfaces. Instead of direct manipulation of icons, GWAIS uses command keys, as is common in EMacs applications.

GWAIS allows users to access the interactive features of WAIS: question entering, relevance feedback, displaying document, and source selection. An extra feature, not found in the other clients, is an interface to an indexer for creating sources. Graphic documents can be displayed on X Windows terminals if the user has set up the environment variables.

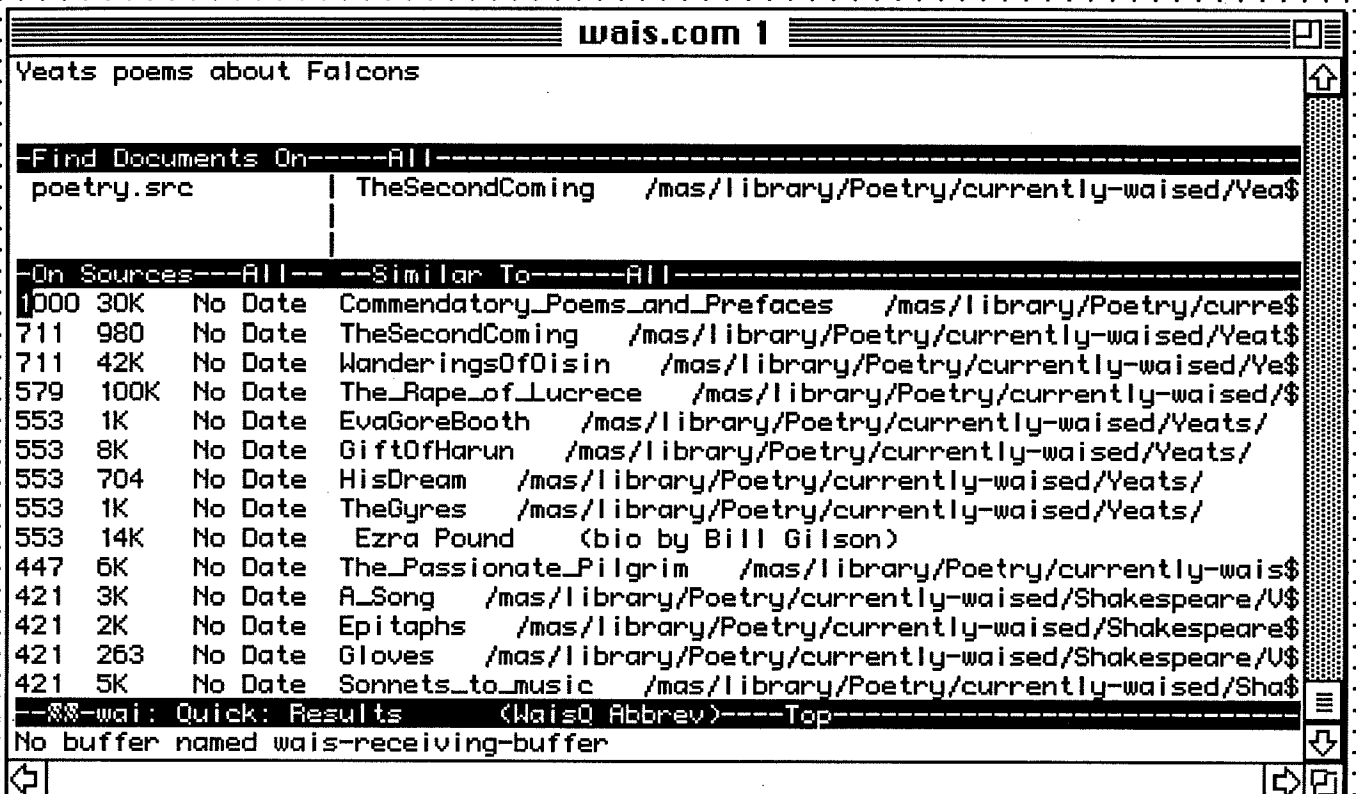


Figure 10. The GWAIS interface, displaying the results of a relevance feedback search.

Screen WAIS (SWAIS)

To open WAIS to a wider community of users, an interface was developed to run on dumb terminals or over Telnet sessions. Screen WAIS, or SWAIS, uses a character display terminal screen for the interface. It is appropriate for dial-up users, Telnet users and low-end terminal users.

SWAIS does away with the step of searching the directory of servers; the interface contains a screen listing all known publicly available servers. There are other screens for displaying search results and retrieved documents. Unlike the other interfaces, the sources list shows what site runs it and how much it costs (if anything). The resulting document screen includes headlines and how many lines the document is, but its innovation is that it shows the source database.

The designers chose to list all servers in order to make it easy to discover servers and to encourage use of multiple databases. Now that there are some 600 servers on the Internet, however, this design has become quite unwieldy and difficult to navigate. What is needed is a customizable screen, like the Sources window in the Mac clients, that allows users to work from a short list of frequently used sources.

SWAIS does not handle relevance feedback or downloading new sources from the directory of servers. And it does have some ease-of-use problems. In "Internet for Dummies," authors John R. Levine and Carol Baroudi write: "Unless you're thrilled at the thought of parallel parking

an 18-wheeler in the city, you're probably not going to be really excited by the standard Unix command-line interface for WAIS."

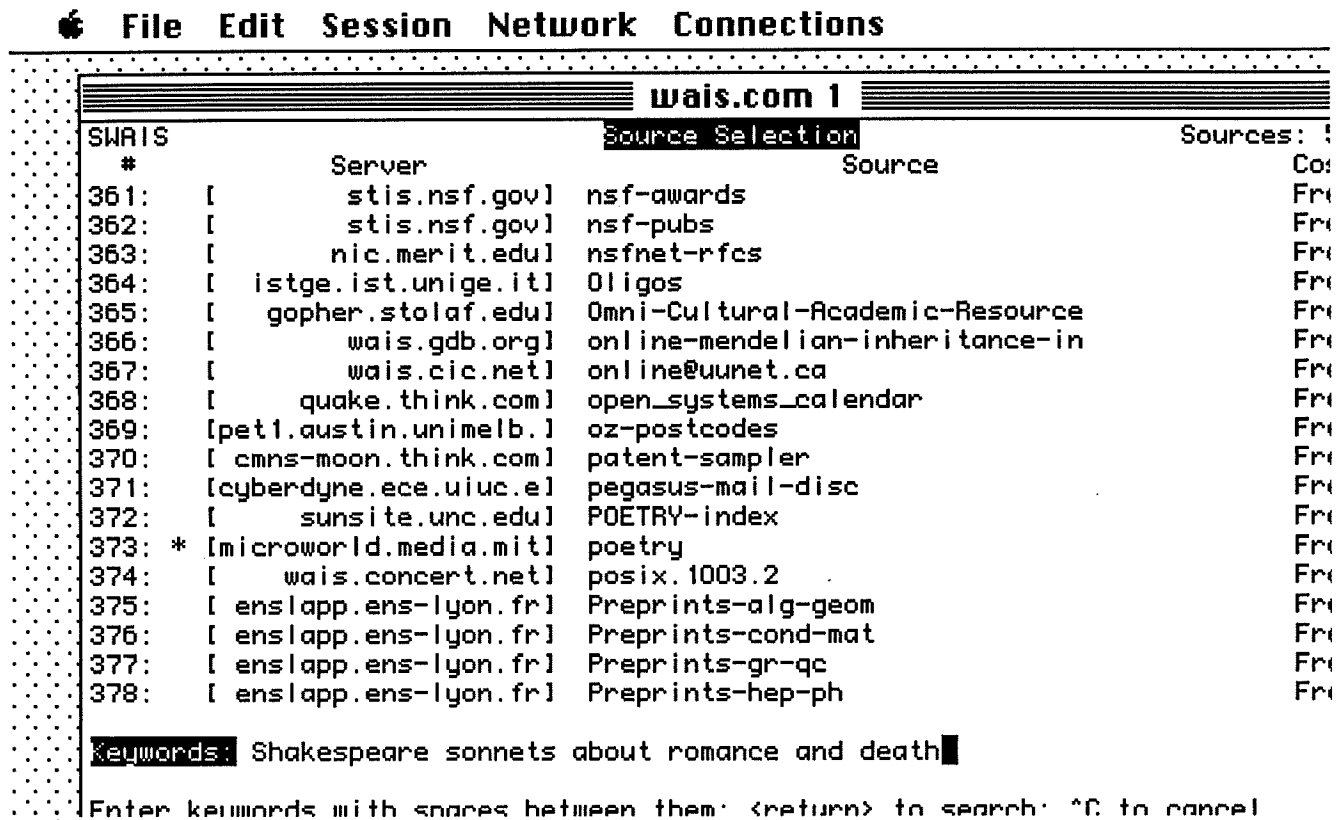


Figure 11 The SWAIS query building screen. The poetry source is selected, and search terms are entered.

Microsoft Windows Interface

Figure 13. WinWAIS query window. ***

WinWAIS was named in 1993 the best Windows application in the government/public administration sector. It is largely similar to the Macintosh clients. One interesting addition is the ability to search for maps based on longitude and latitude. This is primarily useful when searching on USGS map databases since few other WAIS servers contain the spatial information this feature requires.

Like the Macintosh clients, WinWAIS users search for sources through the directory of servers, save them in a Sources window and can query multiple databases. Result ranks are indicated by stars, with four stars being the most relevant and one star the least. Relevance feedback is also supported.

MCC has also published a Windows client with similar operation but no support for spatial searching.

Conclusion

There will be many different network interfaces as the different uses and user bases become more distinct. Page metaphors may persist for researchers, but gaming and entertainment uses may move toward dynamic and picturesque interfaces. Maintaining compatibility at the protocol level will enable the same databases to be used from these various interfaces. Thus the role of the protocols is to keep the systems from flying into separate isolated client-server systems that do not share resources. If commonality in protocols can be maintained, then new interfaces can be written without having to reinvent all the server and protocol infrastructure.

Table 2: Freeware Clients and Locations

Client	Author	FTP Location
DOS	Jim Fullton, UNC	/pub/wais/DOS/*@sunsite.unc.edu, or /pub/tcpip/pcwais.zip@hilbert.wharton.upenn.edu
Gopher		uminn
GWAIS (Gnu Emacs)	Jonathan Goldman Thinking Machines Corp.	/pub/freeware/unix-src/wais-8- *.tar.Z@ftp.wais.com
IBM Mainframe	Tim Gauslin, USGS	/pub/freeware/ibm-mvs/*@ftp.wais.com
Mac	Harry Morris, WAIS Inc Francois Schiettecatte	/pub/freeware/mac/wais-for-mac*@ftp.wais.com /pub/freeware/mac/WAISBrowser*@ftp.wais.com
Mac HyperCard	Francois Schiettecatte	/pub/freeware/mac/HyperWais*@ftp.wais.com /pub/freeware/mac/JFIFBrowser*@ftp.wais.com
Mail	Jonathon Goldman Thinking Machines Corp.	send message to waismail@quake.think.com, "search <source-name> {keywords}" or "retrieve DOCID" (DOCID as returned by a search)
Mosaic	NCSA	????????????????
NeXT	Paul Burchard, Univ of Utah	/pub/freeware/next/*@ftp.wais.com
Openlook	Simon Spero, UNC	/pub/freeware/open-look/*@ftp.wais.com
OS2	Kevin Oliveau, WAIS Inc Julie Mills, Library of Congress	/pub/freeware/os2/*@ftp.wais.com
SunView		/pub/wais/sunview/*@sunsite.unc.edu
SWAIS	John Curran, BBN	/pub/freeware/unix-src/wais-8- *.tar.Z@ftp.wais.com
Telnet Access	(uses SWAIS)	Telnet quake.think.com, login wais, password user@host
VMS	Jim Fullton, UNC	/pub/wais/vms/*@sunsite.unc.edu

Windows	Tim Gauslin, USGS	/pub/freeware/windows/wnwais*.zip@ftp.wais.com
	Kevin Gamiel, MCNC CNIDR	/pub/NIDR.tools/wais/pc/windows@ftp.cnidr.org
XWAIS	Jonathan Goldman Thinking Machines Corp.	/pub/freeware/unix-src/wais-8-*.tar.Z@ftp.wais.com

Trademarks in this chapter: AppleSearch (Apple Computer), Relevant (Ensemble Inc)

CHAPTER 3: THE WAIS SERVER

Introduction

The client program is the interface that allows users to ask questions of servers. The protocol provides the framework that allows questions to be transported over networks. But the brains of the WAIS system -- what determines the quality of answers you get and the kind of questions you can ask -- resides in the server.

Currently there are three major implementations of WAIS server software: the freeware software distributed by Thinking Machines from 1990-1992¹⁴; freeWAIS, a revised freeware server, based on the contributions of many in the WAIS community and distributed by CNIDR¹⁵; and WAIS Server, a commercial product developed by WAIS Inc¹⁶. References to "advanced" or "high-end" servers generally refer to the WAIS Server product; references to public domain servers generally refer to freeWAIS. It should be remembered, however, that the public domain software is distributed with source code and can be customized to offer advanced features as needed. In any case, there is a great deal of similarity in the underlying operation of the programs; the difference is essentially one of advanced feature sets.

¹⁴ The most recent and widely used version is 8b5.

¹⁵ CNIDR information TK

¹⁶ Wide Area Information Services, Inc., 1040 Noel Drive, Menlo Park CA

Becoming a Network Publisher

Introduction

Keep in mind that there are at least two equally important parts to a successful publishing system. One is the quality and value of the information; the other is the ease and effectiveness with which users can find and access the information they want. WAIS technology deals only with the second part of the equation, but because WAIS supports unstructured data in formats including pictures, sound, digital video, spreadsheets, software -- anything that can be stored electronically can be published over the network. No matter the format, WAIS users can navigate through large numbers of documents via text-based search and browse techniques.

The requirements of network publishing are:

- A collection of information that you want to publish,
- The WAIS server software,
- A UNIX computer and disk on which the information resides,
- A TCP/IP network to connect clients to the server,
- An Internet connection for publishing beyond an internal network.

Database

The database of information may consist of free-form text, images and multimedia documents. It does not have to be organized in a structured form, as in traditional relational databases. Furthermore, your data collection can be of any size. It may be as small as one megabyte or as large as 50 gigabytes. The collection of documents exists on a disk directly connected to the Unix computer running the server software.

Server software

WAIS server software creates an index to facilitate fast search and retrieval of your data. This package also contains the search engine software; the server software that handles connections from clients as well as restricting access and monitoring usage; and the WAIS protocol suite for communicating with clients.

The WAIS server software runs on most UNIX platforms from IBM-PC compatible hardware to multiprocessor servers. Since WAIS-compatible clients are available on most platforms and operating systems and the WAIS system uses a protocol based on industry standards, servers can communicate with WAIS-compatible clients regardless of the client's platform, operating system or system configuration.

Hardware

The hardware required to run the WAIS server depends on the size of the database and the number of searches expected. For those expecting very heavy loads with large databases,

benchmarking is recommended. In practice, we have found that mini-computers such as Sun, DEC, and HP machines are fast enough to serve large databases to thousands of users.

TCP/IP

TCP/IP¹⁷ is the Internet protocol, the lowest-common denominator for transporting information between computers. By using this industry-standard protocol, the WAIS system is guaranteed to get information where it's going. TCP/IP gives the server the appearance of a dedicated connection to a client, and guarantees the validity of information passed between them. The WAIS protocol suite sits on top of the TCP/IP protocol, and manages many higher-level functions specific to information publishers and users.

Internet connection

The Internet is an international computer network used by educational, commercial, military and government organizations -- and, increasingly, small businesses and individuals. To date, several million users in dozens of countries use the Internet, which is growing in geometric proportions. The Internet provides a very wide and broad audience for network publishers. Even if your source is not available to the general public, the Internet is an excellent delivery mechanism since WAIS lets you control who has access. Security in WAIS will be discussed in more detail below.

The Internet also delivers an installed base of users; there are tens of thousands of users of free WAIS, Gopher and World Wide Web clients on the Internet.

Network publishing does not require the Internet; if you are publishing information only to networked clients in your organization or over another wide-area networked environment, you won't need the Internet.

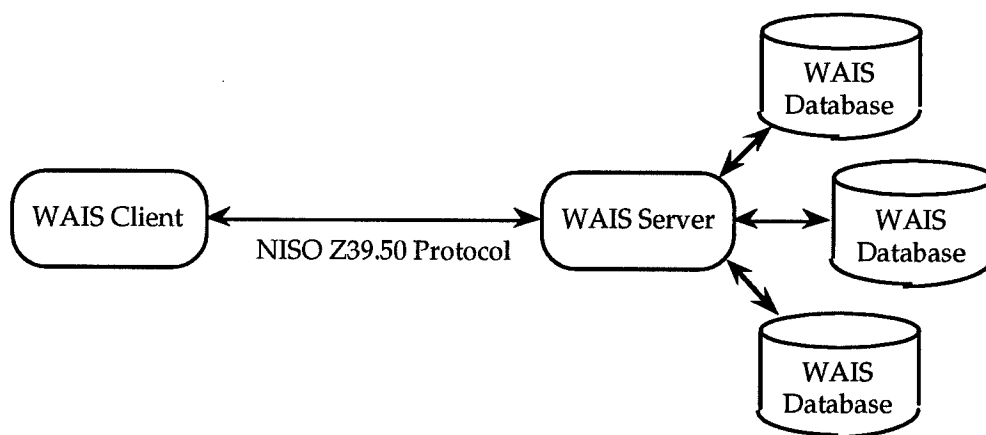


Figure 1: The WAIS Server Architecture

¹⁷ Transmission Control Protocol/Internet Protocol

Inside the WAIS system

There are really two main parts of the server side of the WAIS system: the server software, including the search engine, security and log functions, and the database, including parsing and indexing. The search engine uses the indexing files in the database to find documents that match the words and phrases in the user's question.

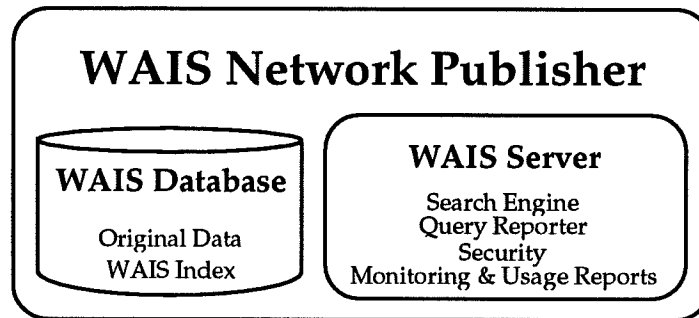


Figure 2: Functions of WAIS database and server

The WAIS Server

The ability to find the 10 to 100 most relevant documents out of the hundreds of thousands of articles in a database is a measure of a good server and has been the subject of decades of research in the field of information retrieval. The WAIS search engine uses some of these statistical techniques to find appropriate documents, while freeing the user from the need to query in a syntax that the computer understands. Statistical techniques used in high-end WAIS servers include phase matching, word weighting, term weighting, among others.

WAIS also takes advantage of relevance feedback searching, a powerful technique that takes advantage of the processing power of the server computer to search for many words in huge collections of information. Relevance feedback lets the user add discovered information to the question, so that many words not initially entered in the question can be included in the search. This form of semantic matching has been very effective for helping users navigate through gigabytes of information. Positive feedback from users gives the server enough information to make more appropriate selections. (See the accompanying article on relevance feedback TO COME).

A Sample Search

In the previous chapter, we showed how users could take advantage of these techniques in searching a server. To understand how the server responds to queries and relevance feedback, let's look at the search from the server's perspective.

The process starts when a client connects to a server by making a question. Before the search transaction takes place, though, the client sends an initialization packet over the network to log in and negotiate packet restrictions. This is invisible to the user but starts the process from the server's point of view.

Perhaps the user is looking for information on tobacco sales in Asia in The Wall Street Journal and other newspapers. The user would type "Tell me about tobacco sales in Asia" into the question field of a WAIS client.

The newspaper server would start a session for the user and log all transactions. At this point the server receives the initialization request and search request with the user's name and Internet address. It would check the *access list* -- a list of machine names that are authorized to use the server -- for that database and either execute the search or respond with a document explaining how to subscribe to the service. In this case the database owner might charge a monthly subscription fee for use of the database. Another for-pay scheme would be to charge users based on the number of searches and retrievals.

Assuming the user has access, the server would use its *search engine* with the database to find relevant documents. Prior to the search, an automatic *indexing* process would have created a set of files that helps speed searching. The engine looks up the queried words in the index's *dictionary*, which contains a sorted list of all the words in the database. The server ignores common words in the question (there are approximately 300 common words like *is*, *and*, *what*, *for*, etc., that are always ignored) and searches for the other words in the dictionary. For each word, the dictionary then points into the *inverted file*, which lists all the occurrences of that word and their locations in the documents. This list is loaded into memory, merged with the lists from other words, and then the documents are ranked in order of relevance.

In this case, the words *tobacco*, *sales* and *Asia* would be searched for, with *tobacco* being the rarest of the three. The inverted file says that *tobacco* is in 571 documents. The other words are in many other documents.

Next, the server consults the *document table*, a listing of all documents, which points to the *headline table*, which in turn points to the *filename table*. Finally the original data file is accessed, the results are ranked in order of importance and returned in a *search response*. The returned results are a list of *citations*, which include the headline, date, length, a document identifier (URL), and the document formats the server supports.

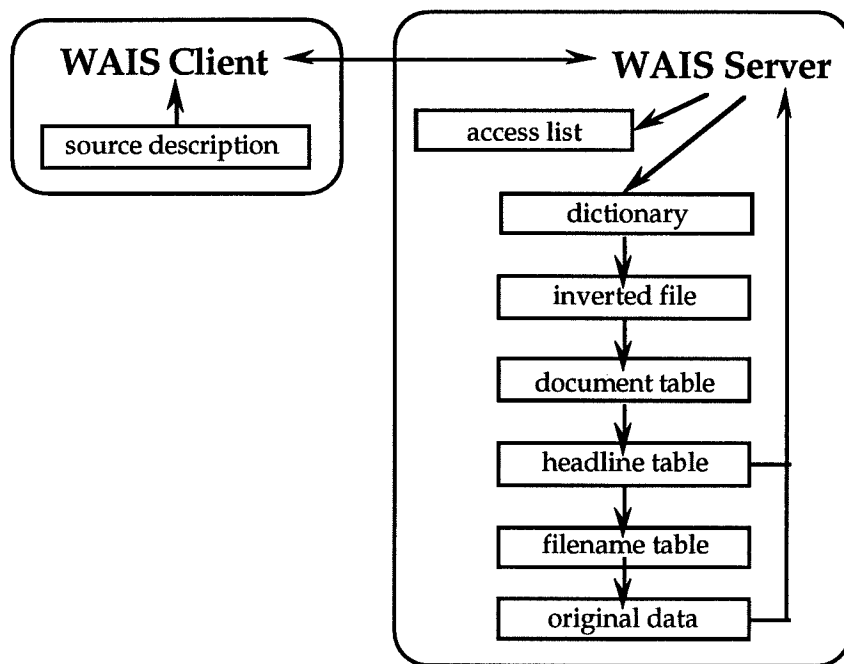


Figure 3: How the WAIS index is used during a search.

The client program would display the list of headlines to the user along with the date and source. By clicking on one of the headlines, the user selects a document to be retrieved.

The server gets a retrieval request with a format the client can accept, for example PostScript. The server then checks again to see that the user has access permission for that document, logs the request and returns the document for the client program to display.

Perhaps the user is especially interested in a paragraph in an article on RJ Reynolds' marketing focus in Hong Kong. The user can say, "I like that, find me more like that one" by selecting the paragraph for relevance feedback. This tells the server to refine the search using those additional words.

The relevance feedback search request from the client includes the document identifier limited to the paragraph with the original query words. This paragraph might contain some place names, factories, or managers' names that would then be used in a search. The resulting large query would help find other documents from the entire database that share the same words and phrases out of the entire database. Care is taken in the search engine to discount words that are not as important; if this doesn't happen, the results can be very misleading.

If the user wanted to eliminate any documents about Korea or only wanted articles from the Wall Street Journal, for instance, the search might be modified to:

"Tell me about tobacco sales in Asia NOT Korea" or "Tell me about tobacco sales in Asia AND source = Wall Street Journal." Best of all would be a combination of techniques; "Tell me about tobacco sales in Asia NOT Korea AND source = Wall Street Journal."

In the final case, only relevant documents from the *Wall Street Journal* that do not refer to Korea would be listed. But the resulting documents are still returned in a relevance-ranked list based on the same criteria as a free-form or "natural language" search. These techniques are

only available in high-end servers and are useful primarily to the trained user. Having a continuum from "user-friendly" to "expert-aware" in one system has proven to be a plus in user acceptance.

It is in this way that users find the documents that are looking for. The server responds to the searches and retrievals typically in under two seconds so that searching is an interactive and iterative process.

In the rest of this chapter, we'll discuss in more detail many of the concepts introduced in the above description.

The Search Engine

The WAIS search engine is at the heart of the WAIS server. It receives a user's question, searches the database for the most relevant documents and returns a ranked list of documents. The standard server implementations include support for natural language questions, relevance ranking and relevance feedback. More advanced implementations offer features like fielded search, Boolean operators, right truncation, stemming, thesaurus lookup, and query reporting.

Natural Language & Relevance Feedback

One of the key features of WAIS is the natural language question format. This means users can simply type in questions or statements expressing their interests. The server does not "understand" the question, per se. Rather, it uses the words and phrases in the question to search for documents that contain those words and phrases. If the words occur many times in a document, it gets more weight, if it appears in the headline it gets more weight, etc.

The chief advantage of natural language is the fact that no syntax need be learned. Anyone can sit down with the program and type in a question regardless if the database being searched is poetry or genetic material.

On the down side, the results of a natural language search are often too inclusive to be very satisfactory. For instance, a search of a business database using the question "Tell me about Apple Computer" might list those about the computer company "Apple" at the top, other hits might come back about the apple industry and several other computer companies. Since there is a limit on the number of hits that can be displayed, the user might get back only a few relevant documents.

So a second searching technique is built into WAIS -- relevance feedback. This function is based on the idea that users frequently do not state exactly what they're looking for in their first question. If one of the hits from the first question turned out to be on a subject closer to the user's aims, he could tell the server to add the words in that document to the search. In this way, the user can feed the server with more and more relevance clues.

(In the example above, if the user was actually interested in executive compensation at Apple, he might use relevance feedback to include a document listing the names of the company's top executives.)

Relevance feedback lets the user select a document or a group of words in a document and add those words to the query. Using the same weighting algorithms, the server determines the most important words and phrases in the relevant document and adds them to the original question. The weight of the relevant document terms is less than the original terms, however, since they were explicitly mentioned by the user. The relevant document terms are not looked up in the thesaurus.

Careful word and phrase weighting is extremely important for relevance feedback since it would be easy to find extraneous documents by following the wrong words. Most free servers, for instance, do not do this weighting very carefully.

Advanced Searching Techniques

Between natural language, relevance ranking and relevance feedback, the WAIS system offers a very easy-to-use interface that often gets users close to information they want. The standard system can be frustrating, however, for users with focused questions and those comfortable with more complex query syntax.

Advanced server software accommodates these users with Boolean operators, fielded search and other techniques familiar to users of proprietary search systems. We'll discuss these below:

Boolean Operators

Using Boolean operators like AND, OR, NOT, and ADJ, users can write much more focused queries than they could with natural language. Their meanings are relatively intuitive:

- AND means that both conditions must be met. In the query "tobacco AND asia," the server is instructed to return only those documents that contain both words.
- OR means that either condition can be met. This is what is assumed in a natural language question. "tobacco OR asia" tells the server to return documents with either word.
- NOT means that a condition must not be met. "tobacco NOT asia" tells the server to return only documents that do contain "tobacco" and do not contain "asia."
- ADJ means that two terms must be side by side. "tobacco ADJ sales" tells the server to return only documents that contain the phrase "tobacco sales" and not documents in which "tobacco" and "sales" are not adjacent.

Boolean operators can be combined in fairly complex ways to voice quite sophisticated queries. "tobacco ADJ sales AND (vietnam OR viet ADJ nam)," for instance, yields documents containing tobacco sales and vietnam, no matter which way it is spelled.

Fielded Search

Fielded search allows more precise searching of semi-structured documents, such as database records or e-mail messages. In fielded search, the regular portions of the documents can be tagged by the WAIS parser as fields. The WAIS Inc. parser can support up to 254 unique fields; public domain servers do not currently support fielded search.

Using fielded search techniques users can search for the presence of text within a field. For example, a researcher might search an archive of his or her e-mail messages for notes about gypsum mines. By issuing the query "subject = gypsum" to a server that supports fielded search, the researcher would obtain a list of all messages with that topic in the subject field. More usefully, fielded search and Boolean operators could be combined for a more directed search. For instance: "subject = gypsum AND Nevada NOT California AND date = November AND 1993" is such a search. In this example, the user is searching for messages from November 1993 about gypsum mining in Nevada but not in California.

Relevance Ranking

No matter which searching techniques were used, the server returns a "hit list", a ranked listing of the most relevant documents. How relevant they really are is something of a subjective call by the user, but this is largely determined by the ways in which the server calculates relevance.

Relevance is scored by the server according to several rules. The most relevant document has the highest score, or rank. A document receives a higher score if the words in the question are in the headline, or if the words appear many times, or if the phrases occur exactly as in the question. A document's score can be derived from using techniques such as word weighting, term weighting, phrase matching, proximity relationships and word density, all of which will be discussed below. However the standard WAIS servers generally use only word density to determine relevance. The other techniques are only implemented in more advanced server products.

Word Density

The ratio of the number of times a queried word appears in a document to the size of the document is called the word density. It is a measure of how important a queried word is to the overall content of the document. A higher word density results in a higher relevance ranking. This is the technique used in public domain servers.

Word Weighting

Each word in a document is given a "weight" that corresponds to how important that word is to the document. The exact weight that a word receives depends on where in the document the word was found. A word is weighted highest if it appears in the headline, less if the word appears in all capital letters or if the first letter of the word is capitalized, and least if it appears only in the text. Servers using only word weighting are dependent on descriptive headlines for best results. (See the section on the database, below, for a discussion of how headlines are generated.)

Term Weighting

This technique differs from word weighting in that it measures how rare a word is in the database. In term weighting, each word used in the database is assigned a numerical value, called the *term weight*, which is based on how frequently that word appears in all the documents in the database. Words that occur in many documents are not weighted as highly as terms that appear in only a few documents. Very common words are either ignored or diminished in the scoring. For example, since the term "animal" may occur frequently in many of the documents in a database, its term-weighting is small compared to a term such as "hippopotamus," which may occur in only a small number of documents.

Phrase Matching

If the user's question contains a natural language expression made up of more than one word, a phrase-matching technique is employed. Using this technique, a higher score is assigned to a document containing a phrase that identically matches the phrase in the user's question. For example: searching for "International Business Machines" would rank documents with those words exactly in that order higher than those documents with those words spread apart.

Proximity Relationships

Proximity relationships designate that if the words in a question are located close together in a document, they are given a higher weight than those found further apart. The idea behind a proximity relationship is that words found in close proximity to each other in a document more likely contain the same content as that specified in the user's question.

Additional Server Features

In addition to searching as outlined above, advanced WAIS servers provide additional features that can be very helpful to both users and administrators. As mentioned earlier, no special clients are required to take advantage of these features.

Customizable Stopwords

A stopword is a frequently used word that, when encountered in a user question, is ignored. For example, since the word "the" commonly appears throughout the English language, it is typically regarded as a stopword. The search engine includes a list of approximately 300 stopwords. Advanced servers allow administrators to customize stopwords, which is valuable for improving the responsiveness of servers and especially for specialized information sources where unusual words may be common.

Right Truncation

Right truncation involves the ability to use wild card characters in a search. As the term implies, the wild cards are only in effect at the ends of words. The wild card tells the server to search on words matching the base characters before the "*" and to ignore any trailing characters. For example, "geo*" would retrieve documents containing the words "geographer," "geography," "geologist," "geometry," "geometrical," etc.

Stemming

Stemming is a technique used to automatically derive variations of a queried word. These variations are then used as part of the search. If a question contains the word "skate," for example, stemming is used to find documents that may also include "skates," "skated," and "skating." High-end servers support two types of stemming -- plural and Porter [Reference?]. In plural stemming, the search engine tries to determine the plural form of a word. In Porter stemming, the search engine attempts to find the real base, or stem, of a word and derive any possible alternate variations. The administrator must select one of these stemming algorithms before indexing the database.

Thesaurus Lookup

Advanced servers support thesaurus lookup to aid the user in finding synonyms for the words in a user's question. Each entry in the thesaurus contains a list of pairs, where each pair is made up of a question and a weight. The question is an expression containing a combination of natural language and Boolean terms, and the weight is a measure of the importance of the pair to the other pairs on the list. The thesaurus is constructed by the database administrator during indexing, and is typically based upon the question behavior patterns of the users.

Since the thesaurus can expand a word into a whole complex query, this feature can be used to implement some of the "concept searching" features in other databases.

Query Reporter

A query report is a document created by the server that describes how a client's question is parsed by the server. When a client asks a question of a database, and if the query reporter is enabled for that server, the server creates and returns a query report to the client. The query report is the last document in the relevance-ranked list of documents returned by the server. The headline of the query report is listed as "Query Report for this Search" and its relevance score is 1. Since the query report is an actual document, it may be retrieved for viewing by the client.

The query report contains the following information:

- The database being questioned,
- The original question,
- The Boolean equivalent of the question in "infix" notation. This notation is a fully parenthesized version of the question, showing the Boolean operator precedence.
- The Boolean equivalent of the question is displayed as a tree.
- The number of documents and the number of words in the database,
- The number of unique words in the database,
- The number of times each word in the question occurred in the database,

- The expanded search words resulting from right truncation, and
- The number of documents found that satisfied the question, and the amount of elapsed time it took to perform the search.

The purpose of this information is to give the user feedback on how the question was interpreted by the server, and on how well the information in the database matched the words in the question. Below is an example query report generated from the following question: "carbon monox* AND poison*." Simply stated, the question is looking for documents on carbon monox* and poison*. The question uses right truncation for monox* and poison* to match words such as monoxide, monoximes, etc., and poisoned, poisoning and poisonous.

Figure 4: Query report

Headline: Query Report for this Search

This is the search report for the search you ran on Jun 3 13:31:51 1993.
It is a temporary file, and will expire about an hour after the search.

Searching /wais/indexes/mydatabase...

Your query:

carbon monox* AND poison*

is equivalent to:

((carbon monox*) AND (poison*))

and was interpreted as:

AND

(carbon monox* poison*)

The database contains 39,062,401 words in 230,750 documents.

There are 639,200 different words.

carbon occurs 30,404 times in 14,896 documents.

monox* is expanded to:

monoxide occurs 3,825 times in 2,515 documents.

monoximes occurs 1 time in 1 document.

monoxodithioacetal occurs 1 time in 1 document.

monooxygenases occurs 2 times in 2 documents.

monoxyhemoglobin occurs 1 time in 1 document.

poison* is expanded to:

poisoned occurs 61 times in 49 documents.

poisoning occurs 486 times in 283 documents.

poisonous occurs 17 times in 14 documents.

The search found 67 documents. It took about 5 seconds.

The search was performed by a WAIS Inc server: WAIS waisserver 1.0.10.
For more information email info@wais.com.

Monitoring and Usage Reports

Besides the work of searching the database and returning answers, high-end WAIS servers automatically record all transactions in a log file. The usage characteristics of your server can be extracted from this log file and summarized in a usage report.

For each client process requesting service, the WAIS server records in the log file the server's process ID, the current count on the number of transactions performed for this client, the date, the time, and the type of transaction. The WAIS server records six main transaction types:

- Opening a connection,
- Searching a database,
- Returning results from a search,
- Retrieving a document,
- Closing a connection,
- Errors and warnings.

If a search transaction is performed, the server also records the name of the database and the client's question. If results were returned from a search, the number of documents found and the document identifiers are also logged. And finally, if a document is retrieved, the document identifier, the database name, the document size, and the document display format are all recorded.

Figure 5: Elements of Usage Report

Total number of connections	The total number of independent client connections made to the WAIS server. A single connection can span over multiple searches and retrievals, and over multiple databases
Number of different machines connecting	The total number of client machines requesting services from this WAIS server.
Total number of searches	The total number of searches requested by all clients.
The total connect time (seconds):	The sum of the connect time of all clients. The connect time is the lifetime of each daemon server process, in seconds. The majority of the connect time is idle time.
The total search time (seconds):	The sum of the search time of each daemon server process, where the search time is the elapsed time, in seconds, that each daemon spends servicing its client's search request.
Searches returning zero hits:	The total number of search requests resulting in no matches, where the daemon server process was unable to find any documents matching the client's question.
Total number of documents retrieved	The total number of documents retrieved by all clients.
Total number of databases searched:	The total number of different WAIS databases that clients have searched.
Number of searches with no DB name	The total number of times clients requested a search without specifying the database name.
Number of searches requesting help:	This number represents the total number of times a client process requested a search for "?" or "help". This gives you an idea of how many new users are requesting information about the databases served on this machine.
Avg. number of seed words per search:	The sum of the number of words contained in all questions divided by the number of questions, or search requests. The word count also includes Boolean operators and stopwords.
Number of searches using relevance feedback	The total number of searches performed with relevance feedback.

Number of server warnings:

The number of times a warning occurred while processing a client's request.

Number of server errors:

The number of times an error occurred while processing a client's request.

The total number of search and retrieval requests for each database searched by a client process. This information gives you a quantitative idea of the load on each database provided by the WAIS server.

The names of all client machines accessing the server's databases and the number of connections requested by each machine.

The names of the client software and the number of connections requested by clients using this software.

The error and warning messages of any problems reported by the server.

TK from Brewster

The WAIS Database

Introduction

A WAIS database is made up of two main components: the collection of information and the indices of this collection. Creating a WAIS source is essentially the process of indexing the words in a database. The set of files created by the indexing process are used by the search engine to provide fast search and retrieval of the original data stored in the database. Taken together, the data and the WAIS index make a complete WAIS database system.

The data consists of a set of documents, headlines and words. A document is the smallest retrievable element of the database. For example, a database may contain a volume of journals, where each article of the journal is a separate document. Another example is electronic mail, where each mail message is a different document. Or a document may be a very small piece of information such as a stock report or a weather report. A document can contain text, images, video, or a combination of types of data. In a database of resumes, for instance, each document might contain the text of the resume, a picture of the person, and even a digital video clip of the individual describing his or her responsibilities.

Each document is associated with a headline. A headline is one or more words that embody the main idea behind the document. This is what users see when a list of hits is returned; it's the only hint they get about the content of the document, so it's important that it be informational. An unintelligible headline, like "12076.txt," is not helpful and the file is unlikely to be downloaded. Generally, each headline is automatically extracted for you from the

database, so the quality of the headline is determined by the parser format. (For detailed information on parser formats, see below.)

Headlines, as well as field information, are located in the documents by a WAIS parser. The parser also identifies the words of a document. When the data is parsed, an indexing program creates a WAIS index. The parser takes the original data and separates it into documents, headlines and words. The indexer takes the information generated by the parser and creates the WAIS index.

The Parser

Introduction

The WAIS parser is a program that separates documents into components consisting of a headline, field information and words. The WAIS parser calls specific routines that handle particular data formats such as email, hypertext pages, etc.

The parser and indexer are two distinct programs. This design creates a more modular and maintainable environment for incrementally developing new parsers without the need to modify the WAIS indexer.

Adding a new parser is as easy as defining a small handful of new functions. In this section we'll discuss the input and output formats and go through the steps required to add a new parser.

Parser Input

As discussed above, the input to the parser is simply a collection of documents. These can be documents of any data type in many different file formats. Additionally, the database may be structured in a wide variety of ways. For example, the database may be organized in a single file, across multiple files, or even as multiple files in different directory trees. The WAIS parser scans the input files and outputs a single common format acceptable to the WAIS indexer.

The WAIS parser needs to know three things: what documents need to be parsed, how the parser should read these documents, and how the client should display these documents.

Specifying which documents to parse is simply a matter of providing a file name or a directory name. When the WAIS parser encounters a new file, it must know how to read the information contained in the file. For example, it must determine if the file contains a single document, or a set of documents. The parser must also be able to distinguish the headline, the field information, and the words of the text. Generally, most files can use one of the parse formats in the table below:

Figure 6: Parse Formats

Dash

Each document is separated by a row of dashes. The line following the dashes is expected to contain a headline, followed by the text of the document.

dvi

Device Independent Printer output files. Filename is used for the headline and the contents of the file supply the words of the document.

filename

Uses filename as the headline. Useful for databases constructed of many individual binary files, for example, the contents of which are not words.

first-line

Specifies that each file contains a single document, and that the first non-blank line of the file is the headline, and the remainder of the file is parsed as words.

first-words

Similar to first-line, except that the headline is the first 100 non-blank characters in the file.

gif

Used for Graphics Interchange Format files. The file is considered to be a single document where the filename is used as the headline, and there are no words in the document. Image files, such as gif, can be associated with a text file and considered as a single document by the WAIS parser, indexer and server.

mail-digest

Used for standard Internet mail digest files. A mail-digest file contains one or more e-mail messages, in which each mail message is parsed as a separate document. The subject line of the mail message is the headline and the body of the message contains the words of the document.

mail-or-rmail

Used for UNIX mail files. It is similar to the *mail* format, except that the sender, the receiver and the date are recognized as field information.

netnews

Used for Internet Network News, where each Network News or Read News file contains one or more news messages. Each news message is parsed as a document, where the subject line is the headline and the body of the message contains the words of the document.

one-line

A simple format that treats each line of a file as a separate document. The line also forms the headline for that document.

paragraph

Like the dash format, this format is useful if a single file contains multiple distinct documents or paragraphs. In the paragraph format, each paragraph is separated by one or more blank lines. The first line of each paragraph is the headline which is followed by the text of the document.

pict

The pict parse format is for Apple PICT image file formats. The file is considered to be a single document where the filename is used as the headline, and there are no words in the document. As with GIF, PICT files can be associated with a text file and considered as a single document by the WAIS parser, indexer and server.

ps

Used for PostScript files. The filename is used for the headline and the contents of the file supply the words of the document.

source

The source file format (.src) is generated by the WAIS indexer for the directory of servers. The file typically contains information about the database, and is parsed exactly like the text file format.

text

In text format, each file is treated as a single document, the filename is used as the headline, and the contents of the file are parsed as words. This format is useful for databases constructed of many individual files. This is the default parse format.

tiff

The tiff parse format is for Tagged Interchange File Formats image files. The file is considered to be a single document where the filename is used as the headline, and there are no words in the document.

Display Formats

The display format determines how a client should display a retrieved document. In many cases the document is a simple text file and has no special display needs. In other cases, the document may be in a specific format that should be displayed with a special display program. For example, if a document was created in Microsoft Word, the client must be told that the document is in Microsoft Word format in order to display it correctly. To do this, a display format of *ms-word* is given to the parser.

The parser associates a display format with each document and passes the display format through to the indexer, which in turn stores it for later use by the server. When a client retrieves a document, the server sends the client both the document and its display format. It is up to the client to decide whether or not it can display this format.

Figure 7: Display Formats

Display Format	Description of Format
DVI	Device-Independent Printer Output
GIF	Graphics Interchange Format
MIME	AT&T Multimedia Document
MS-EXCEL	Microsoft Excel Spreadsheet
MS-POWERPOINT	Microsoft PowerPoint Slides
MS-WORD	Microsoft Word Document
PERSUASION	Aldus Persuasion
PICT	Apple PICT Image
PS	PostScript
QUICKTIME	Apple Quicktime Movie
TEXT	ASCII Text
TEXT-FTP	Special FTP File Format
TIFF	Tagged Interchange File Format
WQST	WAIS Question Format
WSRC	WAIS Source Format

Parser Output

The output of the WAIS parser is a formatted stream that can be piped directly into the WAIS indexer. The basic entity output by the parser is the document. For each document, the parser outputs specific document information: the headline, file information, field information, a date and the words of the document.

Document Information

For each document, the parser outputs document information containing the location of the document relative to other documents in the database, and the names of the file or files that make up the document.

The parser records the location information of each document using a document identification scheme. Each document encountered by the parser is assigned a unique document identification number, or doc-id. The parser also records the doc-id of the containing, or parent, document. These identifying numbers are used to determine if the document is a piece of some larger document, such as a section of an article, or a chapter in a book. The parser also records whether or not a document is one of an ordered series of documents. Taken together,

the doc-id, the parent doc-id, and sequential document ordering make it possible for users to systematically browse through documents in a database. The parent doc-id enables the user to locate the parent document, and the sequential document ordering enables the user to find the next or previous document in a series of documents.

As part of the document information, the parser also extracts and records the name of the file or files that make up the document. Typically there is one primary file containing the text of the document and several secondary files that contain audio or visual information.

<talk about URL's>

Headline

Each document contains a headline -- one or more words specifying the main theme of the document. The headline is used for subsequent indexing, search and retrieval. The parser determines where to extract the headline by knowing the parse format of the file that is specified by the user. For example, if the parse format is specified to be *filename*, the parser extracts the headline from the name of the file.

File Information

Each document is extracted from one or more files. For each file associated with a document, the parser records information about the file. This information includes the date the file was created, the date the file was last modified, the display format of the file, and the location in the file where the document begins.

Date

The parser extracts a date for each document in the database. A date consists of a year, month, day, hour, minute and second component. If a document is made up of several files, the parser records the date of the most important file.

Field Information

A field is one or more words that identify a specific characteristic of a document. For example, a document could have an "author" field whose value is the name of the author of the ensuing document. Each document in the database may have a different value for its author field. The main purpose of the field is to allow a user to restrict a search to only those documents having a specific value of the field.

A document may contain zero, one or more than one field. In addition, a field may be identified by more than one name. In e-mail documents, for example, the "sender" and the "from" fields are different names for the same field. The parser must be able to recognize the name of each field and parse its associated value. The parser then sends this information to the output stream for further processing by the indexer.

Words

If a document contains text, each word in the text is extracted by the parser. The parser outputs each word, the weight of the word, and the location of the word in the file.

Developing A New Parser

When a new database format is encountered, one of two strategies may be used to parse the new format. If possible, the data may be converted to one of the existing WAIS parse formats. When conversion is not a possibility, a new parser can be developed to parse the new format. Due to the modularity of the WAIS parser, developing a new parser is quick and easy. The development involves the addition of up to five new functions and the addition of a new "defparser" data structure.

The WAIS Index

The index is the core of the WAIS database. This is where searching and cross referencing take place. In the previous chapter on clients, we walked through the search process from the user's point of view.

Figure 8 Components of WAIS Index

Source Description	<ul style="list-style-type: none">• Describes the database and the server.• An editable ASCII file, it is used by the client to contact the server and search the database.• Source description files describe the database, the server on which the database physically resides, and the cost of using the database. This information is used by the client software to learn about the database.
Access List	<ul style="list-style-type: none">• Contains the addresses of all machines allowed to search the database. Editable files.
Dictionary	<ul style="list-style-type: none">• Contains a sorted list of all the words used in a database.
Inverted File	<ul style="list-style-type: none">• Contains a list of all the words in the database; for each word it lists all the documents containing that word.• Inverted file is a sorted index of entries, where each entry corresponds to a word in the database, the number of times the word was used in the database, and a pointer to a position in the "Postings" file.
Document Table	<ul style="list-style-type: none">• Contains a record of each document in the database.
Headline Table	<ul style="list-style-type: none">• Contains headlines of all the documents in the database.

Filename Table

- Contains a list of the filenames of the original data files.

Catalog

- Contains a list of headlines and document identifiers for some or all of the documents in the database. This list may be returned to a user whose search has gone poorly, as an aid to help them understand the contents of the database. The catalog file contains the headlines and corresponding document identifiers of all documents in the database.

An added feature in some servers is incremental indexing, which allows the publisher to add more documents to the WAIS database without having to reindex. It indexes only those documents that are new or have changed since the last time the data was indexed. This is an important feature for network publishers whose data changes often or when database size is large.

Other Server Issues

Security

Publishers may want to limit access for security reasons or to enforce a for-pay scheme or for several other reasons. While the public domain WAIS servers offer limited security controls, there are a number of security options, available as third-party additions or through standard computing procedures. These are discussed below:

Access List Security

The WAIS server uses an access-list security system to limit client access to WAIS databases. Before processing a client's request, the server checks to see if the requesting client has access to the requested database. It does this by checking an access list maintained for each database. The access list tells the server the legal client machines that have access to the database. The identity of the machine is based on the machine's Internet address. The access-list security system is available with all existing WAIS clients, and is built into most WAIS server implementations.

Firewall Protection

For secure sites, a "firewall" security system is recommended. In this system there is a single gateway, or firewall, that filters all incoming and sometimes outgoing WAIS packets. A common configuration includes:

- A crypto-key system for outside dial-up and telnet users.
- Restricted or no-access admitted for outside WAIS users.

- Internal WAIS users use a forwarding mechanism to allow access to outside servers.
- Only secure physical access is allowed to the physical firewall machine.

Server Forwarding to Obscure User Profiles

A forwarding server can be used much like the firewall protection method, but for use in obscuring message traffic. This can be done through a variety of heavily used machines so that traffic patterns are difficult to trace.

Authentication Option for the WAIS System

Kerberos-based authentication is a system designed and built by MIT for use on the Internet. MCC, the Austin, Texas-based consortium, offers this as an add-on to WAIS software. This comes with special client software, but uses WAIS server technology. This scheme offers centralized key management for database holdings.

Encryption Option for the WAIS system.

Public key authentication and encryption systems work well in the WAIS system. RSA and PGP2 encryption systems can be added to WAIS since the directory and direct contact architecture is modeled on the Whitfield Diffie Public Key system structure. With this structure, a variety of communications systems can be used without changing the encryption scheme. PGP2 is based on public-domain algorithms, while RSA is based on patented algorithms. (See the accompanying article on public key encryption.) (TO COME)

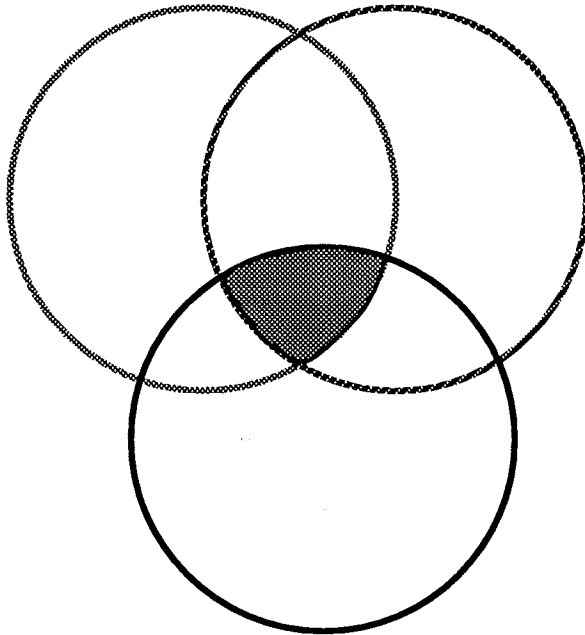
Internationalization of WAIS

TO COME

How does WAIS compare to other search engines?

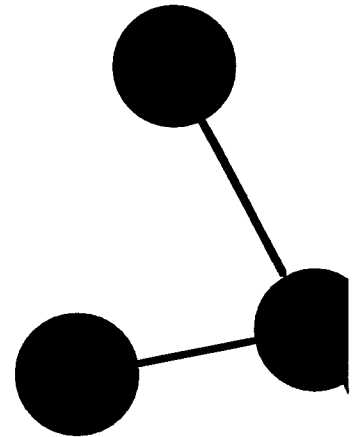
TO COME

Boolean Search



**Retrieve documents containing
specific combinations of words**

Conce



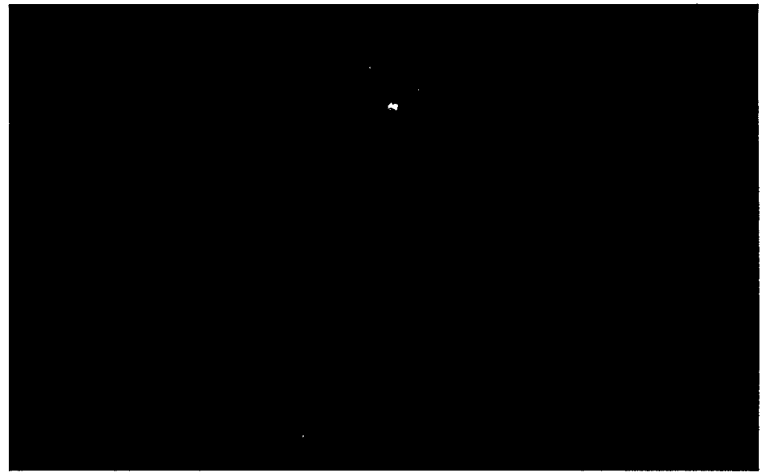
**Explore
containi**

Boolean Query

Hard to Use:
Complex Syntax



Poor Results:
The wrong information
No ranking of results

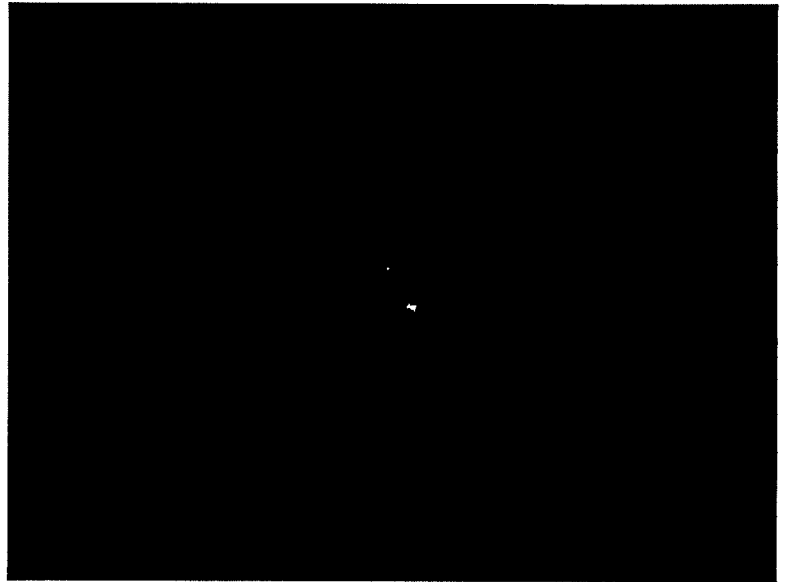


Conceptual Search: Phase

Easy to Use:
No Syntax



Options:
**What do you want
to follow up?**



Conceptual Search: Phase

Relevance Feedback:

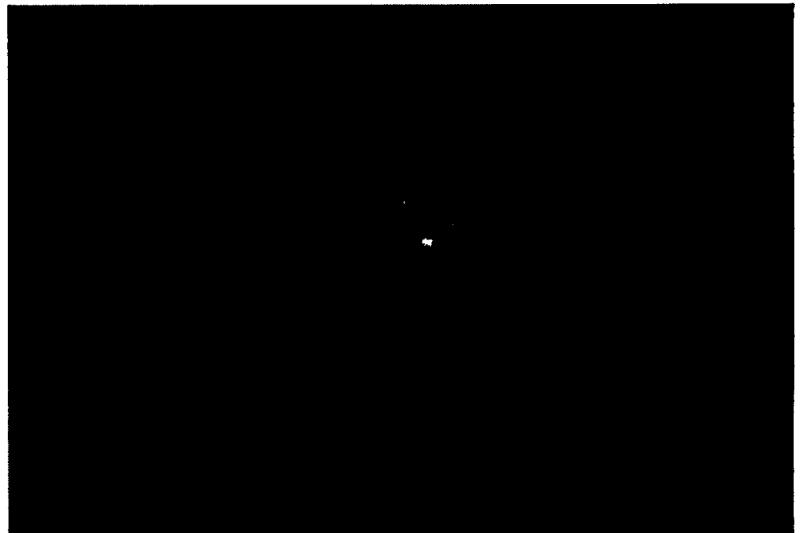
**I like these; show
me more.**

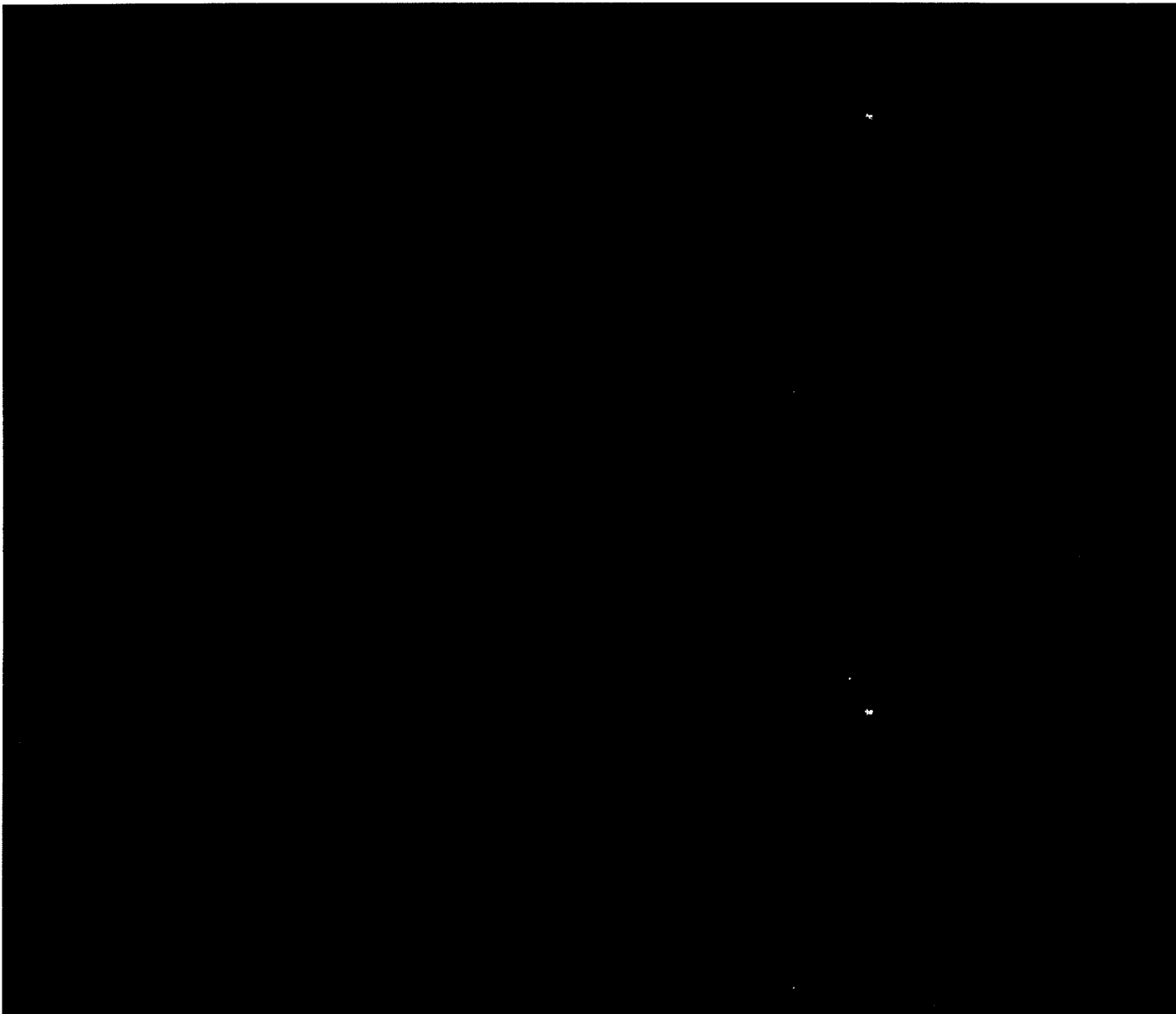


Improved results:

**Articles on related
topics are found.**

Results are ranked.





THE WAIS PROTOCOL SUITE: A BASIS OF NETWORK PUBLISHING

"... they have all one language; and this is only the beginning of what they will do; and nothing that they propose to do will now be impossible for them."

Genesis 11:6

The thousands of network publishers and millions of readers on the Internet are connected by network publishing protocols. They make up the common language that allows users to find servers and pay for access, search and browse databases, and retrieve multimedia documents.

Network publishing protocols are application protocols that allow multiple computers to work together to form coherent applications that take advantage of wide area networks, specialized client and server programs, and cross-vendor compatibility.

The importance of protocols is perhaps best illustrated by another publishing industry computer protocol -- PostScript. While not usually thought of in these terms, PostScript is in fact a protocol for moving a page from a computer to a printer. Adobe Systems (Mountain View, CA) describes PostScript as a page description language that provides device-independence printing of fully composed pages.

The rise of PostScript and desktop computers in the publishing industry led to the decline of host-based proprietary pagination systems like the old Atex systems. With PostScript in place as a protocol, a large and varied industry developed, offering hardware peripherals and innovative software based on PostScript.

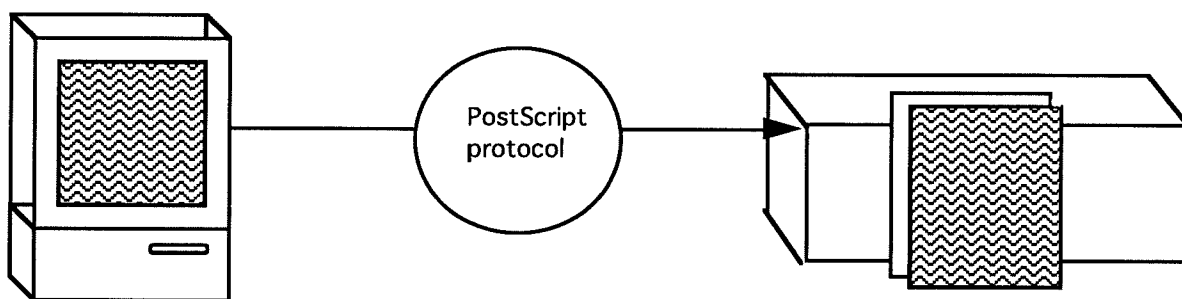


Figure 1. Desktop publishing systems use the PostScript protocol to communicate the layout of a page from a personal computer to a printer.

Similarly, network publishing protocols are moving online services away from the host-based systems like CompuServe and America Online that most people are familiar with and towards an open client/server environment in which applications and services span many computers on different continents. Network publishing protocols allow users to find appropriate servers,

search and browse through databases located on many different computers, and retrieve multimedia documents.

Network publishing protocols are some of the first interactive protocols operating on the Internet; others will follow to serve different industries. And as with PostScript, the acceptance of industry protocol is fostering the growth of new companies, which are providing new specialized clients, servers, development tools and other related products.

There are already three major applications for navigating through information on the Internet. The WAIS, World Wide Web and Gopher systems have proven that users have different navigation needs. The existence of all three has resulted in a rich environment for browsing and searching through information resources. Over time, the protocols for these applications will evolve towards each other, and eventually the capabilities of all three will be available in an integrated way. In the meantime, it's important that they all communicate and interrelate.

In this chapter, we will consider the WAIS protocol suite as a network publishing protocol. We'll consider the requirements of such a protocol, walk through a sample protocol session, give an overview of the WAIS protocol suite and discuss future directions for the protocol.

Requirements of a Network Publishing Protocol

A protocol for network publishing must satisfy several needs, which will be discussed below. Primarily it must provide full-text searching and browsing capabilities across many remote systems. While WAIS system is not the only system to do this; other search standards include Boolean searching over host-based systems (such as Dialog and Mead), CD-RDX¹⁸ (designed for CD-ROM-based databases), NSQL¹⁹ (which extends the Structured Query Language to search full text), World Wide Web/HTTP²⁰ (designed for hypertext browsing over the Internet) and Gopher (hierarchical browsing over the Internet).

But searching functionality is not enough for a network publishing standard. Such a protocol must also offer the following values: It must be functional, scalable, based on open systems, flexible and it must work with other information technologies. In this section, we will discuss why each of these attributes is required and how the WAIS protocol suite delivers them.

Functionality

Since users of the system will work on all different computer platforms and represent a wide spectrum of searching sophistication, the protocol should support all users' queries and return

¹⁸ Silver Platter Information, Inc Norwood Massachusetts.

¹⁹ Fulcrum Technologies, Inc. Ottawa, Canada.

²⁰ Cite CERN, Berners-Lee?

information in many data types. As discussed elsewhere, the WAIS protocol supports natural language queries, which the most novice searchers can write, as well as a variety of advanced search techniques that allow advanced users to issue highly targeted queries.

For responses, the protocol supports multiple data types, so users receive file in formats that their client programs can display. The protocol must also support multiple languages.

One of the biggest obstacles for users is finding the right server since they do not know beforehand what servers are available, how to contact them, or how much they cost. The WAIS system provides a mechanism for finding appropriate servers through the source description file, a form that contains information about the contents of the database. Currently these source descriptions are indexed in another database called a directory of servers.

Scalability

To support a worldwide network publishing system, the protocol must be remarkably scalable. It must accommodate quite substantial growth in the number of servers involved in the system. There are currently about 600 WAIS servers on the Internet. The protocol should be able to accommodate 6,000 or 60,000 such servers.

These servers may contain databases of greatly different sizes. Something like a phone book may be less than a megabyte, while a database of, say, all patent applications would be tens of gigabytes. As the system grows and more information is published on the network, most databases will grow in size. Again, the protocol must be scalable to very large databases and support databases of very different sizes on the same server. The WAIS structure will support a system serving up many terabytes of information.

Open Standards

The WAIS protocol suite is based on national and international standards for communicating between clients and servers since this offers both technical and market-growth advantages. As different vendors offer compatible systems for serving different markets, the protocol becomes the important piece for tying the implementations into a working whole. The WAIS protocol enables the interoperability of a global system made up of thousands of interacting pieces.

The alternative to basing the system on open standards is to use proprietary standards. While proprietary protocols may be more responsive to user needs because the authority to change them is centralized, the proprietary nature places reliance on a single company. Network publishing depends on a "critical mass" of publishers and users. Avoiding impediments to achieving widespread acceptance is essential to the long-term success of the system.

Allow for Billing and Security

A commercial network publishing system, of course, needs mechanisms for billing and security. Exactly what billing method will be used is not clear. Among the options are pay-per-minute, subscriptions, pay-per-search and pay-per-retrieval.

With the WAIS protocol, any one or a combination can be used, since the server can log user activity. The protocol provides a mechanism to restrict access for security and charging. Any

user might be able to get information about a database but not able to use it; in other cases no information should be known to outside users. Varying levels of security must be available on a database-by-database level to satisfy concerns about security and charging.

Flexible to Adapt to the Changing Needs of a New Industry

A protocol must be flexible enough to adapt to the emerging needs of this new industry. Both new and old technologies -- such as video, wireless networks, palmtop computers and home game machines -- may have a role in a network publishing system of the future. The protocol must be able to accommodate all these and be flexible enough to meet circumstances that we cannot even imagine. This means that the standards communities behind the different pieces must be responsive and a mechanism for introducing new standards into the suite is required.

Interoperable with Online Systems

In order to interoperate with other online systems, application-level gateways have offered a solution. A wide variety of interfaces -- including Gopher, WWW, e-mail, as well as independent online services like America Online -- can gateway to the WAIS protocol because it makes no assumptions about the presentation of the output.

The WAIS protocol suite is also flexible enough to run on any reliable digital transport. On the Internet this is TCP/IP. But it could run over X.25, ISDN, Novell, AppleTalk or other transport protocols. Because interactive searches work at relatively slow speeds, 9600 baud is adequate for text retrievals, while 56KB is more appropriate for graphics and other non-text media.

Certainly, as the network publishing system grows, so will the protocol. Limitations will be uncovered. Additional capabilities will be added. But because the WAIS protocol suite meets the above criteria, particularly in terms of openness and flexibility, it will evolve to meet the needs of an constantly changing environment. What network publishing will look like a year or two from now, we cannot predict. But the WAIS protocol suite should be flexible enough to continue as the system's backbone.

Protocol Session

Perhaps the best way to understand the role of the protocol is to look at what it does during a search-and-retrieval session. There are three phases to a session: initialization, search and retrieval.

Initialization Request and Response

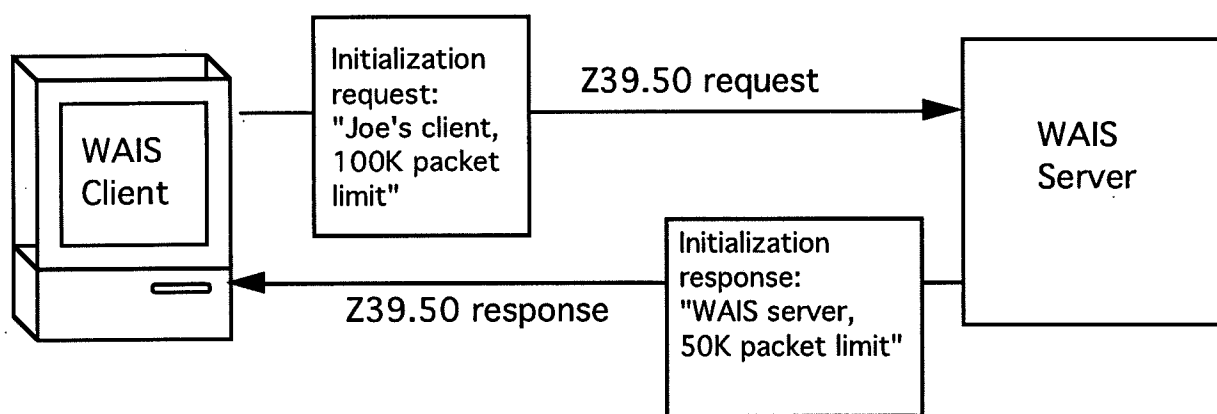


Figure 3: The first stage of a session is initialization. Typical number of bytes is 100.

The initialization stage is a simple transaction in which client and server establish identity and capabilities. The initialization request from the client states who the user is, what client version is being used, packet limitations and protocol version. (Establishing identity is a complex procedure, which is dealt with separately in the Billing and Security chapter.)

Based on this information, the server responds with information about packet size, server version or diagnostic information.

Search Request and Response

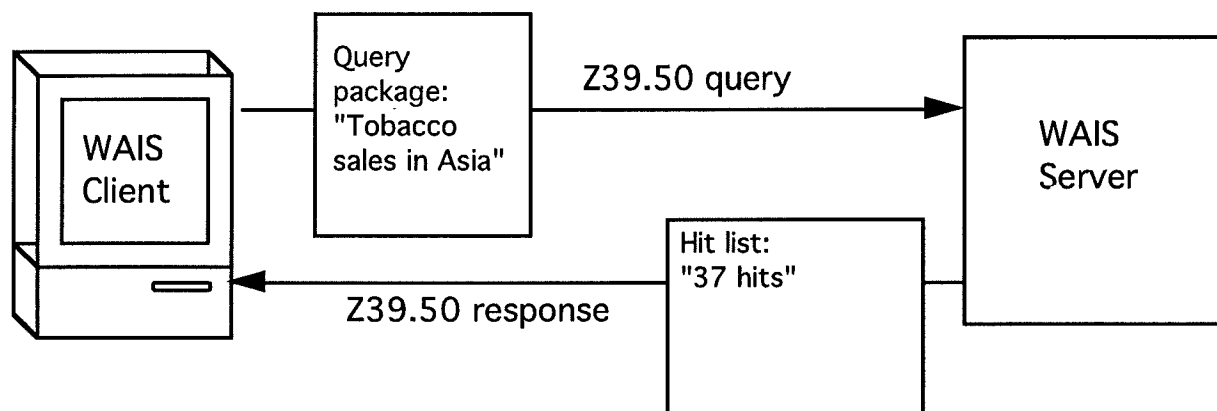


Figure 4: In the search stage, the client asks a question and the server returns a hit list.

The search request (100-200 bytes) is a bundled-up version of the query, including query words, relevant documents (when relevance feedback is used), what databases to search and how many hits to give back on first response.

The search response (about 3,000 bytes) is a sequence of records, citing the list of hits, ranked in order of relevance. These citations are:

- headline (80 character description of the document or picture)
- date (using ANSI date format: YYMMDDHHMMSS)
- document identifier (using IETF standard Uniform Resource Locator (cite))
- list of types (using MIME [Multipurpose Internet Mail Extensions]²² types or other type designators)
- best section (the byte that starts the "best" section of the document)
- score (an absolute score from 1 to 1000 of how well the document matches the query).

The number of hits returned can be limited by packet size, maximum number of hits specified by the client, or the number of documents deemed appropriate by the server. The client can then request the next group of hits in the same search set.

When there are no matching documents, some implementations of the protocol return help and catalog documents. When access is denied, servers can return appropriate information.

Retrieval Request and Response

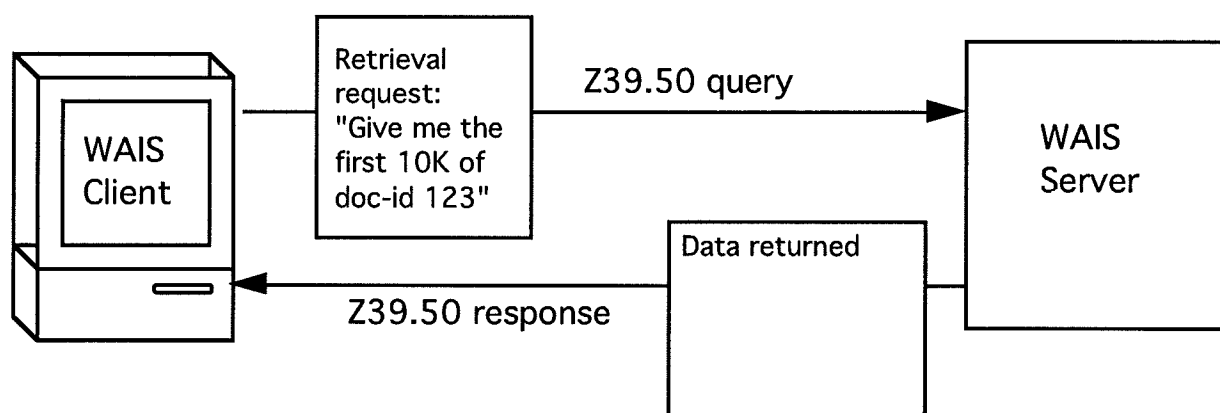


Figure 5: In the retrieval stage, the server provides the requested document in the way specified by the client.

When a user wants to retrieve a specific document, the client sends a retrieval request, including:

- document identifier (can be a URL or an index into the result set)

²²N. Borenstein, N. Freed, RFC 1341, "MIME: Mechanisms for Specifying and Describing the Format of Internet Message Bodies."

- document format (e.g., HTML, JPEG, MS Word, etc.)
- unit of retrieval (e.g., byte, line, frame, etc.)
- numeric range (e.g., 0-1000).

The client can retrieve a directory by searching on the associated document identifier.

The server's retrieval response is either a bunch of bytes or a diagnostic. Long documents -- whether text or other media -- are retrieved in segments.

Interoperability on the Internet

Applications like Gopher and Web work with WAIS by creating gateways between systems, effectively becoming WAIS clients. Most gateways, however, are not as fully functional as dedicated clients.

Getting from Gopher to WAIS Servers

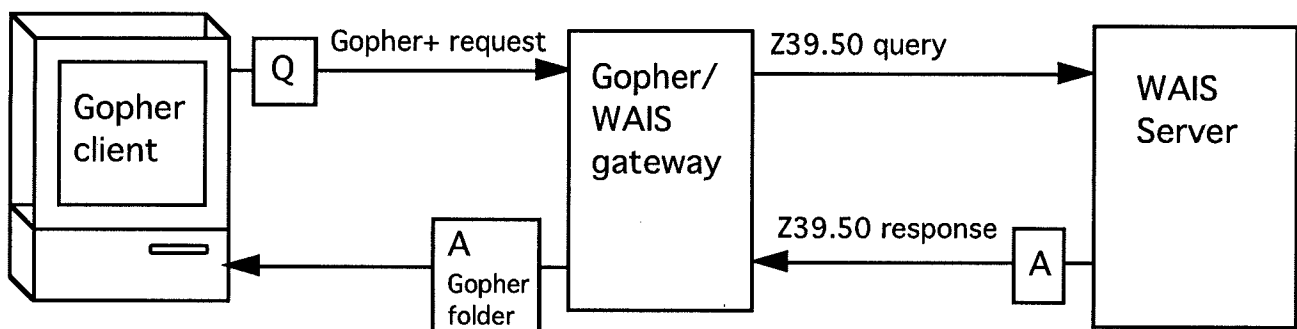


Figure 6: Gopher users can query WAIS servers by means of a gateway that packages Gopher+ requests into Z39.50 format.

Gopher is a client-server hierarchical browsing tool in which servers present menu items from which clients choose to navigate through sets of files. Gopher users can also search through Gopher or WAIS databases. This is done by having the Gopher server search the WAIS database either via gateway to the WAIS protocol or by directly searching the WAIS index files. This section will describe the two methods.

The gateway method works with all WAIS servers and permits links to any WAIS server on the Internet. In effect, the Gopher server acts as a WAIS client to obtain information. These servers might be remote WAIS servers or simply a WAIS server that has indexed the contents of a Gopher hierarchy.

The gateway is created by compiling WAIS protocol code into Gopher so that it can emit queries. Then a directory for WAIS source files is created in the Gopher hierarchy. Finally,

pointers are entered for the source files. Each entry has a line describing the name, path, port number, type, and hostname of the link.

In the example below, the directory of WAIS sources contains a file called "movie-reviews.src," which is a standard WAIS .src file. The server for this database is at "mit.edu" on port 210. All Gopher-to-WAIS links are type 7 links. The Gopher menu item which describes this link will say "Movie Reviews From Usenet." Here is the sample entry:

Name=Movie Reviews From Usenet

Path=waissrc:/.waissrc/movie-reviews.src

Port=210

Type=7

Host=mit.edu

In another method, one can make Gopher hierarchies -- or, in fact, any files on the server -- available for WAIS searching with a Gopher client by creating a WAIS index of those files. To do this, the search code from the WAIS protocol must be linked into the Gopher server.

Work is currently being done to allow a WAIS server (or any server that accepts Z39.50 input) to search a Gopher hierarchy.

Getting from World Wide Web to WAIS Servers

The World Wide Web (WWW) is a means for authoring multimedia documents, publishing them on the Internet and navigating from one document to another across the network by hypertext links. See the Clients chapter for a more detailed of the Web.

The Web consists of thousands of hypertext document servers, which communicate using HyperText Transfer Protocol (HTTP). Most of these documents are in a standard format called HyperText Mark-up Language (HTML), which is a special subset of the Standard Generalized Mark-up Language (SGML). The hypertext pointer used in HTTP is described by the Universal Resource Locator (URL) standard. .

A Web client, such as Mosaic, communicates with a WAIS server in one of two ways. In one method, the client directly queries a WAIS server using the WAIS protocol. Alternatively, the client communicates using HTTP with a remote machine that runs a gateway, which in turn queries a WAIS server. Both methods of accessing WAIS databases are described below.

Direct query

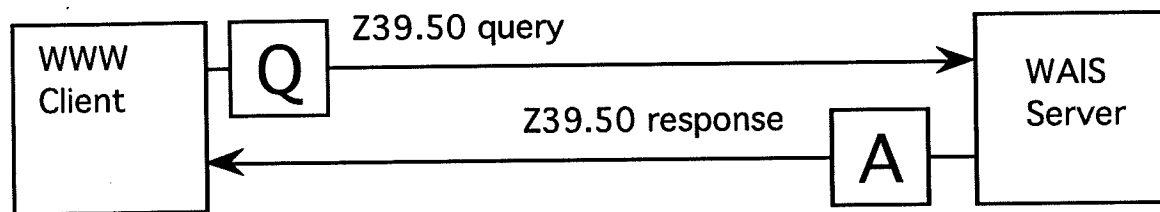


Figure 7: Direct method for Web clients to query WAIS servers.

The client program acts as a Z39.50 client to search WAIS databases and retrieve documents. Web clients generally format the Z39.50 response into a hypertext page, with each headline appearing as a hypertext item. WAIS URLs underlying each item point to the document. Clicking on the item retrieves the document.

HTTP-to-WAIS Gateway

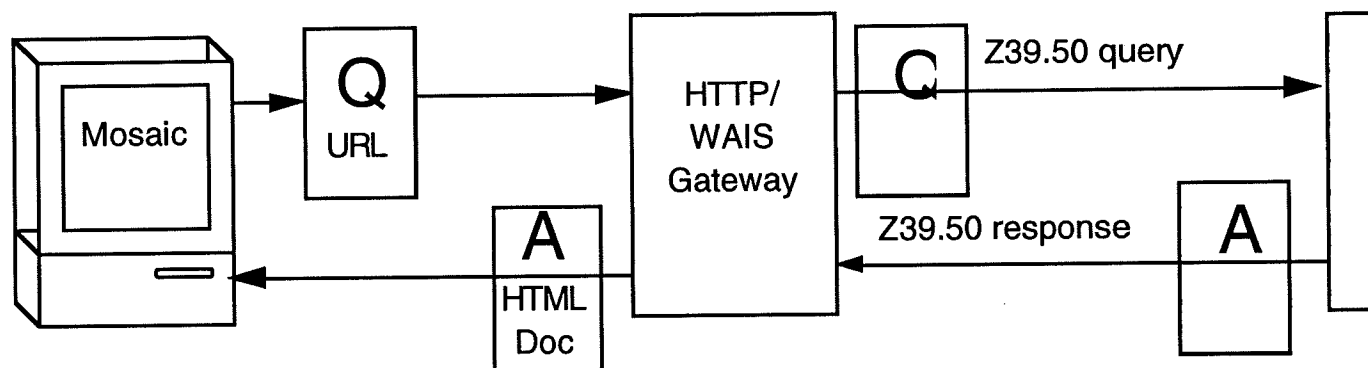


Figure 8: Using a gateway, the client sends a query as a URL and receives a hit list in the form of a hypertext document.

In a gateway environment, the client interacts with the gateway the same way it does with any HTTP server -- by supplying a URL and retrieving a document. The client encodes the query in an HTTP URL, which is sent to the gateway. The gateway translates the HTTP URL into a Z39.50 request and sends it to the WAIS server. When the WAIS server sends back its reply, the gateway translates that reply into an HTML document to return to the client.

An example of such a gateway is WAISgate by WAIS Inc. It does two things: translates .src file into an html page, and provides the gateway service when that page is in use. This page can have forms for fielded search as well as natural language and Boolean searching, if the server supports these options.

Searching HTML Documents with WAIS

For documents that originate as HTML pages, the WAIS indexer can index documents and serve them to World Wide Web clients, which retrieve them as HTML pages. In this application, Z39.50 is usually used only to communicate between processes within one machine.

How WAIS works with Online Services

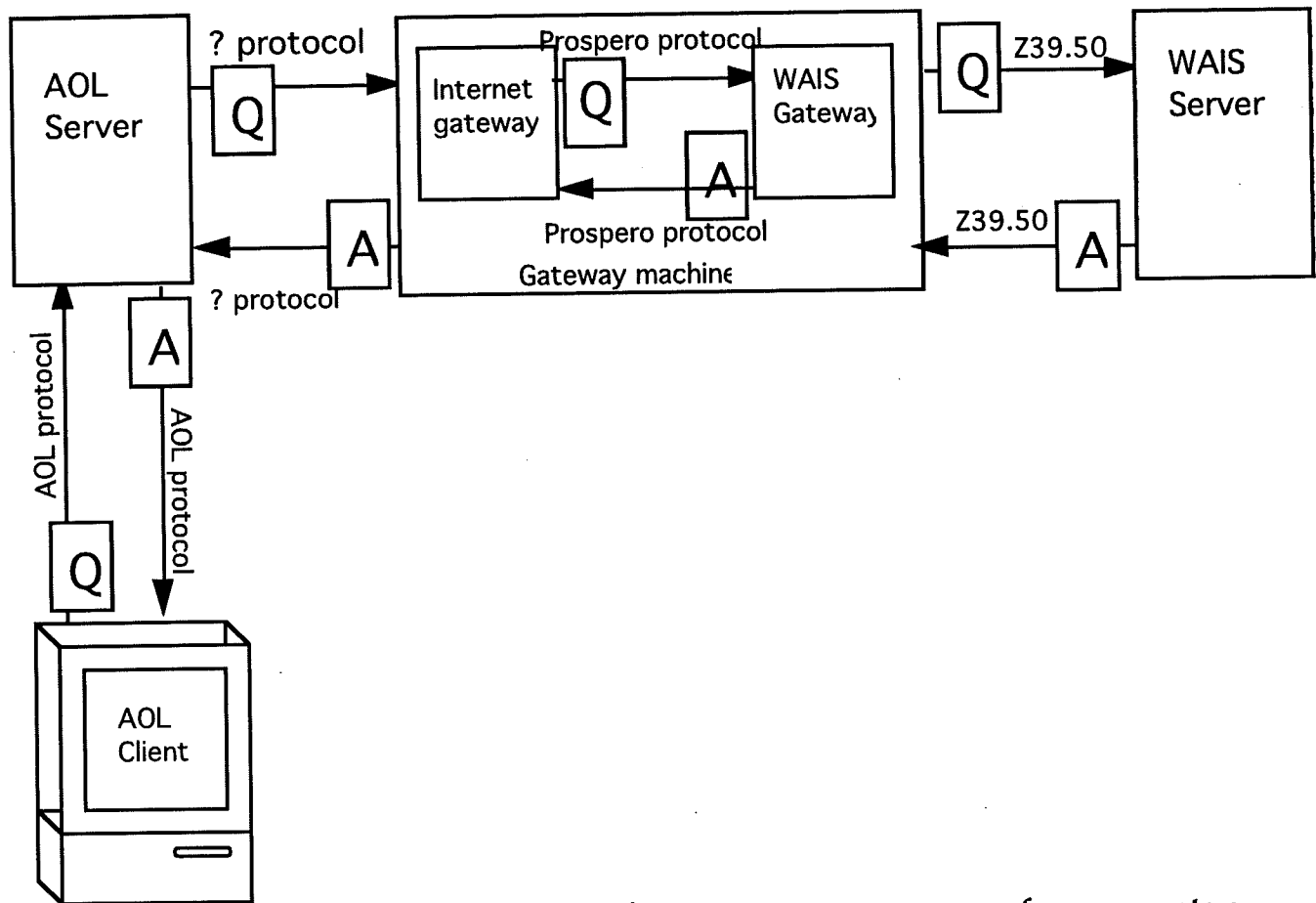


Figure 9: America Online uses a number of gateways to move a query from a user to a WAIS server.

Numerous online services with dial-up users are using gateways to provide Internet services to users. The first service was email, which is now widely available from most services. Now Netnews, WAIS and Gopher services are starting to be made available. This use of gateways to Internet resources offers a wider audience to Internet publishers than any single service could offer. The online services provide value by providing easy access for users and established billing infrastructures.

America Online, for example, is providing access to Gopher and WAIS databases through a gateway. Here the America Online protocol is used between the client and the server, which talks to an Internet gateway using the Prospero protocol, which in turn talks to Gopher and WAIS gateways. The Prospero protocol offers a single method to access multiple resources in an efficient way.

Given the interest in the Internet, it is likely that gateway access to information services will become as commonplace as e-mail.

Overview of the Protocol Suite

The WAIS protocol suite is a collection of several different standards working together to provide a standard for network publishing. These standards -- which include Z39.50 and the WAIS profile -- are woven together in the WAIS RFC²³ document, which specifies how they interrelate and define compliance with the WAIS system.

For example, many document formats are supported since many existing document formats are in current use and are likely to change. Support for multiple standards also lessens reliance on any one proprietary standard. This prevents customers from being locked into a single vendor's proprietary system.

This section describes the various parts of the WAIS protocol suite.

Table 1: Evolution of Z39.50

Z39.50-v1	Z39.50-v2 with WAIS RFC
Boolean search	Natural language and Boolean
Bibliographic Records	Many media (text, graphics, sound, movies, etc.)
Query parsed on client	Query parsed on server
No relevance feedback capabilities	Relevance feedback documents included in query
Unlimited number of result sets, which remain on server until end of session.	Stateless server; result set not stored by the server.
Retrieval by indexing into result sets.	Uses URLs.
Requires OSI Reference Model	Runs over TCP/IP, X.25, etc.
Retrieve whole records*	Retrieve documents in pieces
One format per document	Multiple formats per document
Return unordered results	Return ranked and scored results

Information Retrieval Standard Z39.50

Z39.50²⁴ is the standard that formed the basis of the WAIS protocol suite. It came out of the library community and was designed to handle bibliographic records. The WAIS protocol suite uses Z39.50 to carry requests and responses between clients and servers, but it doesn't define what the requests or responses look like or how they're formatted. In other words,

²³RFC #, is an informational RFC with Internet Engineering Task Force. location on wais.com

²⁴ ANSI (American National Standards Institute) NISO (National Information Standards Organization) Z39.50 Information Retrieval Service and Protocol Standard. The national standard is a superset of the ISO (International Standards Organization) Search and Retrieve Service Definition and Protocol Specification standard (ISO 10162 and 10163).

Z39.50 is the boat carrying cargo between two ports. The decisions about what goes into the boat are made by the various other standards that are part of the WAIS protocol suite.

Z39.50 is currently in Version 2. The next version is due in 1994. These successive drafts have wrapped in what has been learned from the experience of use on the Internet.

Version 2 makes several improvements over Version 1. It allows the client to ask for successively greater numbers of citations in response to a query. This means the server is not stateless, as in Version 1, but experience shows that the added functionality is necessary. Version 2 also adds the capability to call out to several other standards, such as URLs and MIME types. It supports browsing, hypertext functionality, authentication and encryption.

In the original WAIS project, the participants (Apple Computer, Thinking Machines, Dow Jones, KPMG Peat Marwick) decided to base the project on an international standard. Z39.50 was selected based with conversations with Clifford Lynch (a leader of the Z39.50 committee) and the fact that the standard was close to what was needed and the standards committee would be flexible to adapting to commercial requirements.

Z39.50 was designed for use by librarians to search remote online card catalogs. Thinking Machines developed a number of extensions²⁵ to the protocol, which supported multi-media data, commercial servers and use by end-users. These extension were incorporated in subsequent versions of the Z39.50 standard.

An implementation of the new WAIS protocol suite was developed by Thinking Machines and distributed on the Internet in the public domain. This "freeware" was incorporated in the development of many information retrieval products and led to the development of 12 clients and six servers in less than a year.

Table 2: Organizations using Original WAIS Protocol Implementation

Organization	Product
Apple Computer	AppleSearch
Johns Hopkins University	HyperWAIS
MCC	WAIS for Mac
National Center for Supercomputer Applications	Mosaic
NearNet	SWAIS
P. Burchard, University of Utah	WAISStation for NeXT
Thinking Machines Corp.	Seeker
University of Minnesota	Gopher
University of North Carolina	WinWAIS
US Geological Survey	DinoWAIS, WinWAIS, SQL gateway
WAIS Inc.	WAIS Server for Unix

²⁵ cite extensions (Franklin Davis, TMC; RFC)

The Document Formats Standard

The protocol can transport any file format from server to client. The server lists the available file types and the client can choose one to retrieve. For example, word processor documents can be integrated into the same system with DBMS records and news feeds. Some formats are SGML, GIF, MARC, ASCII, Microsoft Word, spreadsheets and CAD drawings.

A server can advertise a document in many formats in order to be compatible with large numbers of client programs. There is no need to store numerous versions of the same document if the server can translate between formats on the fly. The most widely available registry of file types is the MIME type standard, maintained by IANA (Internet Assigned Numbers Authority). But the WAIS protocol can use any string of characters as a type, which is important for ad hoc or special-purpose file types.

Different aspects of the document can also be listed as types. For instance, abstracts, copyright information and digital signatures can all be associated with a single document.

Document Identifiers

The server returns a list of hypertext document identifiers as the result of a search. Hypertext has text that includes pieces of other documents and points to other documents. A hypertext pointer is the reference to a related document. The standard for hypertext pointers is the URL, which gives the location and retrieval method. URLs allow a user to disconnect from the server and retrieve the document at some later time.

This structure also allows one document to refer to another without having to republish it. This eliminates copyright violations by always pointing back to the original source and allowing the original owner to control access and be paid for the document. It's easy for users to locate and access remote documents in this way.

The advantages are that popular documents can be tracked and therefore kept current. However, problems occur when documents are moved, deleted, renamed or become obsolete. Ongoing work on the URL standard may resolve these problems by establishing repositories and directory structures for finding deleted documents.

Server Descriptions for the Directory of Servers

A server is described in a set record structure called a source description file. This file contains information on how to contact the server, cost and a description of the server, who maintains it and other information. It is an ad hoc standard defined by a specification by Thinking Machines.²⁶

²⁶ The specification can be obtained at wais.com:ftp/pub/protocol/WAIS-Source-Description.txt

```
(:source
:version 3
>window-geometry (:rect :left 20 :top 74 :right 455 :bottom 389 )
:ip-name "quake.think.com"
:ip-address "192.31.181.1"
:tcp-port 210
:database-name "directory-of-servers"
:cost 0.000000
:cost-unit :dollars-per-minute
:maintainer "jonathan@think.com"
:description "This is the main directory of servers. It is simply a WAIS database which
contains descriptions of other servers. You can use these descriptions to
contact servers around the world.
```

To get a feel for the directory of servers try a search like:

```
'what servers are available over tcp/ip"
)
```

These source descriptions are used by other services to contact WAIS servers. Users can retrieve source files by querying "help." Servers with multiple databases have an INFO database, an index of the source files on the server. Most importantly, they are indexed into several "directories of servers," so users can discover appropriate servers. Thinking Machines maintains a directory of servers for all publicly available servers, while many organizations have specialized subsets. For instance, NASA maintains a directory of servers that lists databases of interest to that community.

Z39.50 Version 3, which is currently in committee, will include some of the functionality of the server description in the "explain" facility.

Other Component Standards

The WAIS profile defines the use of Z39.50 Version 2. It was passed by OSI Implementors Working Group (OIW).

The Z39.50-over-TCP/IP profile is an OIW profile that defines how Z39.50 Version 2 is to be used over TCP/IP.

Z39.53-199X Language Codes is a NISO (National Information Standards Organization) draft standard that defines almost 400 three-character alphabetic codes to indicate language in the interchange of information.

The Generic Record Syntax is an extensible standard for communicating the results of a search. This structure holds the headline, date and author information.

Evolving Aspects of the Protocol

New pieces that are being added include:

Authentication and Encryption

Authentication and encryption facilities are needed in highly secure environments. Password systems, unfortunately do not scale well into a client-server environment since a user would have to remember a large number of passwords, one for each server. The Kerberos system from MIT and public-key encryption systems offer a more scalable solution for the heterogeneous network environments available today. MCC has integrated Kerberos and the WAIS freeware server with their EINet system.

Usage Log Files and Summaries

WAIS, Web and Gopher servers all generate activity logs that show what computer used the server, what they looked for and what they actually downloaded. These logs can be useful for refining a server and for billing purposes.

Log files are created as the server is running and there is no standard format for representing usage data in files, so each server implementation uses its own log file format. These files tend to present a lot of information.

Log file summaries are also ad hoc in format and will evolve to fill particular needs, such as billing, surveys, etc.

These logs open basic questions about how this data should be handled. What should be done with this data? What should users expect will happen? These issues are discussed further in the Billings & Security chapter.

Billing Format Standards

Billing information is derived from activity logs. Currently, there are no standard tools for extracting this information, so individual server administrators go through the logs and create the necessary reports to make bills. Ideally, servers would feed billing information directly into standardized billing reports. One possible standard is EDI (Electronic Data Interchange, ANSI X-12), a file format for expressing forms. There are separate forms standards for communicating billing information across networks. EDI is not commonly used on the Internet.

Query Formats

The Z39.50 protocol represents a query as a nested tree structure if the client parses the query. In the WAIS profile, however, clients do not have to parse queries; they can pass user input directly to the server. This is advantageous because different servers may require queries to be parsed in different ways; it would be undesirable to require every client to know how to talk to every kind of server. This strategy allows for experimentation in query formats and, of course, makes clients easier to write.

Conventions are emerging for representing common query types. The following types are commonly used in the WAIS system:

- Natural language (e.g., "Tell me about tobacco sales in Asia.") This is the default case that all servers answer to an unformatted string. If the server does not know how to parse the string any other way, it assumes a natural language query.
- Boolean (e.g., "tobacco AND Asia NOT Japan") Certain special words when in upper case are used as Boolean operators, such as AND, OR, NOT and ADJ. Parentheses are also used to guide parsing. Some high-end servers understand not only Boolean queries but also mixed natural language and Boolean.
- Fielded search (e.g., "author = John Doe") In a semi-structured database where records contain fields, queries can specify a search for text within a field. Field names are designated by the server on a database-by-database basis. Fielded search can be combined with both natural language and Boolean.
- Spatial searching (e.g., "****") The US Geological Survey is pursuing standards for searching areas of the globe using latitude and longitude and returning maps.

There will be other special cases for searching, such as DNA searching, which will require query formats, first as conventions and later possibly as standards.

Hardcopy Delivery of Documents

Without widespread page image standards for computers, there is a growing demand for fax and other hardcopy delivery facilities. For example, the Z39.50 Implementors Group is now actively discussing mechanisms for requesting fax delivery of documents.

Submitting Documents to a WAIS Database

WAIS servers are currently read-only but it would be useful if users could submit documents to databases. Standards for this are currently being addressed.

Foreign Language Character Sets

As the the Internet spans more countries, smoothly searching across languages becomes important. But problems remain in the conventions that are used even in these languages. For example, how are the Spanish ñ or the French accent marks to be typed on an English keyboard? How should a server interpret these characters or their absence?

There are two major approaches, both with problems, to passing foreign language character sets in the protocol. One is to transmit all queries in Unicode; the other is to express it as an array of bytes and say what character set it's being transmitted in. These issues are currently being discussed.

Future Directions of the Protocol

The primary goal for future development of the WAIS protocol suite is to facilitate for-fee and for-free publishing over networks. This would include such technologies as wireless communication, video browsing, structured databases, palmtop computers and home game machines.

A different approach to protocols is typified by PostScript and TeleScript, protocols that send code instead of a command or a form across a network. This kind of protocol offers intriguing possibilities for searching remote information sources. For example, TeleScript facilitates alerting, in which "searching agents" are planted on servers and send findings back to the user.

CHAPTER 5: WAIS SYSTEMS

CHAPTER 6: FUTURE DIRECTIONS

ACKNOWLEDGEMENTS

INDEX